1243

NASA CONTRACTOR REPORT

NASA CR-2410

NASA CR-2410

NOLIN - A NONLINEAR LAMINATE ANALYSIS PROGRAM

John J. Kibler

Prepared by
MATERIALS SCIENCES CORPORATION
Blue Bell, Penn. 19422
for Langley Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • FEBRUARY 1975

Field	Value
1. Report No.	NASA CR-2410
2. Government Accession No.	
3. Recipient's Catalog No.	
4. Title and Subtitle	NOLIN A NONLINEAR LAMINATE ANALYSIS PROGRAM
5. Report Date	February 1975
6. Performing Organization Code	
7. Author(s)	John J. Kibler
8. Performing Organization Report No.	
9. Performing Organization Name and Address	Materials Sciences Corporation Blue Bell Office Campus Merion Towle Bldg. Blue Bell, Pa. 19422
10. Work Unit No.	
11. Contract or Grant No.	NAS1-12492/126/CRD
12. Sponsoring Agency Name and Address	National Aeronautics and Space Administration Washington, D.C. 20546
13. Type of Report and Period Covered	CONTRACTOR REPORT
14. Sponsoring Agency Code	
15. Supplementary Notes	FINAL REPORT

16. Abstract

A nonlinear, plane-stress, laminate analysis program, NOLIN, has been developed which accounts for laminae nonlinearity under inplane shear and transverse extensional stress. The program determines the nonlinear stress-strain behavior of symmetric laminates subjected to any combination of inplane shear and biaxial extensional loadings. The program has the ability to treat different stress-strain behavior in tension and compression, and predicts laminate failure using any or all of maximum stress, maximum strain, and quadratic interaction failure criteria.

This document provides a brief description of the program including discussion of the flow of information and details of the input required. Sample problems and a complete listing of the program is also provided.

Field	Value
17. Key Words (Suggested by Author(s)) (STAR category underlined)	COMPOSITES, Laminates, Nonlinear Analysis
18. Distribution Statement	Unlimited STAR Category 32
19. Security Classif. (of this report)	Unclassified
20. Security Classif. (of this page)	Unclassified
21. No. of Pages	100
22. Price*	$4.75

*For sale by the National Technical Information Service, Springfield, Virginia 22151

TABLE OF CONTENTS

NOLIN

A NONLINEAR LAMINATE ANALYSIS PROGRAM

by

John J. Kibler

SUMMARY

A nonlinear, plane-stress, laminate analysis program, NOLIN, has been developed which accounts for laminae nonlinearity under inplane shear and transverse extensional stress. The program determines the nonlinear stress-strain behavior of symmetric laminates subjected to any combination of inplane shear and biaxial extensional loadings. The program has the ability to treat different stress-strain behavior in tension and compression, and predicts laminate failure using any or all of maximum stress, maximum strain, and quadratic interaction failure criteria.

A second program, UNI, has been developed which computes elastic constants and thermal coefficients of expansion for laminae, from constituent properties, to aid in compiling input for the NOLIN program. Laminae properties can be computed for isotropic or transversely isotropic fibers in an isotropic matrix. In addition, the nonlinear inplane shear stress-strain curves are computed for the laminae by computing the Ramberg-Osgood shear stress parameter.

This document provides brief descriptions of both programs, a description of the flow of information through the NOLIN program, and detailed descriptions of the input required for each program. Sections are provided with sample problems and sample program output, along with complete listings of each program.

1. NOLIN Program Description

1.1 Introduction

The NOLIN program codifies a nonlinear, plane-stress, laminate analysis wherein the nonlinear behavior of the laminae under inplane shear and transverse extensional stresses are taken into account. Both the underlying analytical development and the computer program are sufficiently general to enable the user to study the nonlinear behavior of a symmetric laminate subjected to any combination of inplane shear and biaxial extensional loadings. Contained in this document are a detailed description of the input required to use the NOLIN program, as well as a description of the theoretical developments upon which the program is based. For a complete theoretical description the reader is referred to Ref. 1.

In unidirectional, fiber-reinforced laminae, the transverse extensional and, particularly, the inplane shear stress-strain relationships cannot be accurately characterized as linear. The NOLIN program allows for nonlinear representations by permitting these stress-strain relationships to take the form of Ramberg-Osgood nonlinear relationships. The introduction of these nonlinear, Ramberg-Osgood type constitutive relationships into a laminate analysis then leads to a set of nonlinear equations involving the laminae stress components as unknowns. The program then solves this set of equations by means of a generalized, Newton-Raphson procedure to give the laminae stresses and strains corresponding to the applied boundary stresses.

The theoretical development for this nonlinear, laminate analysis incorporates total deformation theory with Ramberg-Osgood type stress-strain characterizations to formulate the governing nonlinear equations. At the outset the compliance tensor is assumed to be the sum of two tensors, the components of one are the usual components associated with linear,

orthotropic, plane-stress elasticity theory, while the second tensor contains the nonlinear elements. By assuming a quadratic interaction of the stress components, and requiring the constitutive relationship to reduce to the relationships for the uniaxial stress cases of inplane shear and transverse extension, the elements of the nonlinear compliance tensor are explicitly determined.

Having the nonlinear laminae constitutive relations, the usual methods of laminate theory are then utilized to obtain the governing, nonlinear equations for the laminate. As in linear, laminate theory, the strains of the individual laminae are first rotated to a common set of laminate axis, and the laminate compatibility relations requiring the corresponding strains of the individual laminae to be equal are then employed. In addition, equilibrium at the laminate boundaries is envoked. In this way the required number of equations involving the unknown laminae stresses are formulated.

The program solution procedure for the set of nonlinear equations involving the laminae stress components is a Newton-Raphson technique generalized to accomodate systems of equations. The starting point for the solution procedure is taken as the solution of the associated, linear laminate problem, where the associated linear problem is obtained by ignoring all nonlinear terms.

Incorporated into the program are three different failure criteria, maximum stress, maximum strain and a quadratic interaction criteria. Any or all of these may be employed by the program user. Unfortunately, the nonlinear aspects of this program preclude the generation of strength envelopes since linear extrapolation is not valid here. Instead a sequence of combined loadings may be run.

The governing equations are formulated so that the three stress components in each lamina are the unknowns. Thus, for an N-layered laminate, the problem is formulated in terms of 3N unknowns. To obtain solutions, 3N equations are then required, and these equations consist of three equilibrium equations and 3(N-1) compatibility equations satisfying strain compatibility between adjacent laminae. The three equations of equilibrium for a laminate under a combined state of stress are,

$$\begin{aligned} \sum_{k=1}^{N} \sigma_{11}^{(k)} t_k &= N_{11} \\ \sum_{k=1}^{N} \sigma_{22}^{(k)} t_k &= N_{22} \\ \sum_{k=1}^{N} \sigma_{12}^{(k)} t_k &= N_{12} \end{aligned} \tag{6}$$

Where N_{11}, N_{22} and N_{12} are the applied stress resultants, t_k the thickness of the kth lamina, and subscripts 1 and 2 denote the laminate axes. The 3(N-1) equations of strain compatibility are,

$$\begin{aligned} \varepsilon_{11}^{(k)} &= \varepsilon_{11}^{(k-1)} \\ \varepsilon_{22}^{(k)} &= \varepsilon_{22}^{(k-1)} \\ \varepsilon_{12}^{(k)} &= \varepsilon_{12}^{(k-1)} \end{aligned} \tag{7}$$

$$k = 2, 3, \ldots, N$$

Equations (6) and (7) are the 3N equations required for the solution of the nonlinear laminate problem. When the stress-strain relations given by equations (2), (4) and (5) are transformed to the laminate reference axes and substituted into equations (7), the governing equations can be expressed

in functional form as,

$$F_k \; (\sigma_1, \; \sigma_2, \; \ldots, \; \sigma_1{}^2, \; \ldots) = 0 \qquad (8)$$
$$k = 1, \; 2, \; \ldots, \; 3N$$

1.3 Method of Solution

Solutions of equations (8) for the 3N stress components are obtained by employing a Newton-Raphson iterative scheme. The functions F_k are first expanded in Taylor series about an approximate set of initial stresses, $\overset{\circ}{\sigma}_j$. Considering only the first order terms of these series,

$$F_k = F_k^{\circ} + \left(\frac{\partial F_k}{\partial \sigma_j}\right)\Bigg|_{\sigma_j^{\circ}} \cdot \Delta \; \sigma_j \qquad (9)$$
$$j, \; k = 1, \; 2, \; \ldots, \; 3N$$

By writing,

$$\Delta \; \sigma_j = \sigma_j - \sigma_j^{\circ}$$

where σ_j are the solution values, equations (9) can be rewritten to give

$$\sigma_j = \sigma_j^{\circ} - \left(\frac{\partial F_k}{\partial \sigma_j}\right) \cdot F_k^{\circ} \qquad (10)$$
$$j, \; k = 1, \; 2, \; \ldots, \; 3N$$

For clarity, the notation in equations (10) is, in expanded form,

$$\sigma_j = \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \cdot \\ \cdot \\ \cdot \\ \sigma_N \end{bmatrix} \qquad (11)$$

$$\sigma_j^\circ = \begin{bmatrix} \sigma_1^\circ \\ \sigma_2^\circ \\ \cdot \\ \cdot \\ \cdot \\ \sigma_N^\circ \end{bmatrix} \tag{12}$$

$$F_k^\circ = \begin{bmatrix} F_1^\circ \\ F_2^\circ \\ \cdot \\ \cdot \\ \cdot \\ F_N^\circ \end{bmatrix} \tag{13}$$

and,

$$\left(\frac{\partial F_k}{\partial \sigma_j}\right) \sigma_j^\circ = \left| \begin{matrix} \frac{\partial F_1}{\partial \sigma_1} & \frac{\partial F_1}{\partial \sigma_2} & \cdots \\ \frac{\partial F_2}{\partial \sigma_1} & \frac{\partial F_2}{\partial \sigma_2} & \cdots \\ \cdot & & \\ \cdot & & \\ \cdot & & \end{matrix} \right|_{\sigma_j = \sigma_j^\circ} \tag{14}$$

The solution for σ_j in equation (10) may be taken as the approximate, initial stress values for the next iteration step, and this process repeated until a result is obtained within some desired accuracy. After the stresses are obtained and transformed to the laminae natural axes, the corresponding laminae strains are determined from equations (1), (2) and (5).

1.4 Computer Program

The flow chart for the computer program is shown in Fig. 1. The major sections of the program are the formation of the governing equations, the Newton-Raphson solution procedure and the failure checks.

Using the computer program notation, the governing equations take the form,

$$[A] \cdot \overline{SG} + \overline{B} = \overline{SGO} \tag{15}$$

Where $\overline{SG}$ and $\overline{SGO}$ are the stress solution vector and the applied stress vector, respectively. A is a matrix of constant elements which are the coefficients of the linear terms in the solution, and $\overline{B}$ is a vector containing the nonlinear terms in the solution. The set of equations (15) are equivalent to equations (8). If in equation (15) the vector $\overline{B}$ is set to zero, the resulting equation

$$[A] \cdot \overline{SG} = \overline{SGO} \tag{16}$$

is the linear laminate solution. The stress vector, $\overline{SG}$, as determined from equation (16) is taken as the initial approximation for the stress vector in the Newton-Raphson procedure.

For the Newton-Raphson procedure it is necessary to formulate the derivative of $([A] \cdot \overline{SG} + \overline{B})$ in equation (15) with respect to σ_j as well as the vector

$$\overline{DC} = ([A] \cdot \overline{SGO} + \overline{B} - \overline{SGO}) \tag{17}$$

The vector $\overline{DC}$ corresponds to the vector $\overline{F_f^o}$ in equation (10), and an explicit evaluation of $\overline{DC}$ is obtained by using the current, approximate value for the solution stress vector, $\overline{SC}$. The derivative of $([A] \cdot \overline{SG} + \overline{B})$ is designated $\overline{DB}$ in the computer program, and is equivalent to the matrix $(\partial F_k/\partial \sigma_j)\ \sigma_j^o$ in equation (10). An explicit evaluation of $\overline{DB}$ is also obtained by using the current, approximate value for the solution stress vector, $\overline{SG}$.

In the program, the external loading is applied in increments. The approximate solution stress vector for the first load increment and the first Newton-Raphson iteration is determined from equation (16). For the second and third load increments, the approximate solution stress vectors for the first iteration are taken as the final solution stress vectors from the previous increments. Solutions for subsequent load increments are initiated by the following algorithm:

$$(SG_{i+1})_{INITIAL} = (SG_i)_{FINAL} * (FACTOR)$$

$$(FACTOR) = \frac{i\ (i-2)}{(i-1)^2} \quad \frac{(SG_i)_{FINAL} - (SG_{i-1})_{FINAL}}{(SG_i)_{FINAL}} \tag{18}$$

The convergence and divergence criteria employed in the program are contained in the following expressions:

$$\left| (SG_{i+1} - SG_i)/SG_i \right| \leq \varepsilon$$

$$\left| (SG_{i+1} - SG_i)/SG_i \right| < \lambda \tag{19}$$

Where SG_i and SG_{i+1} are the solution vectors obtained from the i^{th} and $i+1^{th}$ iterations. Usually values of 10^{-3} and 10^{+4} are taken for ε and λ, respectively. However, the other values may be input as data to the program. In addition, the maximum number of iterations to be allowed is input as data. Ten iterations have been found to be sufficient for most problems.

The program contains three failure criteria, maximum strain, maximum stress, and a quadratic interaction criteria. After a solution is obtained for each load increment any or all of these failure criterias may be applied to check for laminae failure.

The maximum stress and maximum strain failure criteria check, respectively, the laminae stress or strain values in

the fiber, and transverse fiber directions against the material allowables. These allowables are input to the program as data. The quadratic criteria is given by

$$A_{11}\ \sigma_{LL}^{\ 2} + A_{22}\ \sigma_{TT}^{\ 2} + A_{44}\ \sigma_{LT}^{\ 2}$$
$$+A_{12}\ \sigma_{LL}\ \sigma_{TT} + B_1\ \sigma_{LL} + B_2\ \sigma_{TT} = 1 \qquad (20)$$

where the coefficients are functions of the allowable stress

$$A_{11} = \frac{1}{F_L^{\ T} F_L^{\ c}} \qquad B_{11} = \frac{1}{F_L^{\ T}} - \frac{1}{F_L^{\ c}}$$
$$A_{22} = \frac{1}{F_T^{\ T} F_T^{\ c}} \qquad B_{22} = \frac{1}{F_T^{\ T}} - \frac{1}{F_T^{\ c}} \qquad (21)$$
$$A_{44} = \frac{1}{(F^s)^2}$$

F_L^t and F_L^c are the allowable tension and compression stresses in the longitudinal direction, F_T^t and F_T^c are the allowable tension and compression stresses in the transverse direction, and F^s is the allowable shear stress. The coefficient A_{12} is input as data to the program, or a default null value is used.

If a failure criteria is satisfied at the end of a load increment, the program determines the failure load through linear interpolation. If all failure criteria are being checked, and not all indicate failure during the same load increment, the program continues loading until all criteria indicate failure.

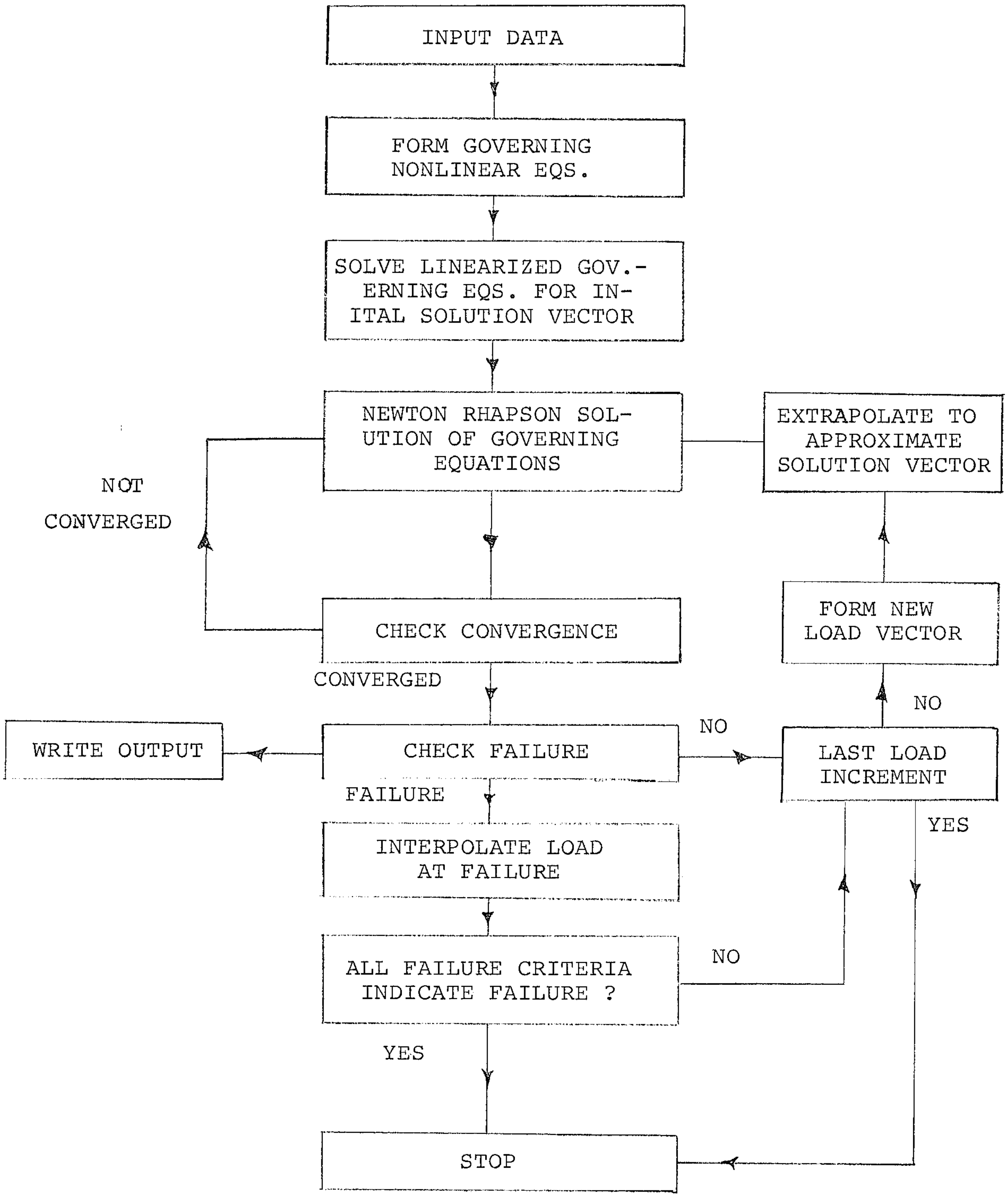

Figure 1 - Solution flow Chart

2. NOLIN PROGRAM USERS GUIDE

2.1 Program Description

This section describes the input data requirements for NOLIN (version 2 mod 2). Input procedures have been streamlined wherever possible by using one NAMELIST statement, hence eliminating any input under format control.

The input flow diagram and input description provide all information necessary to specify input data sets capable of exercising all program options.

2.2 Input Description

The initial data required is a message of five cards of alphanumeric descriptive information describing the problem being solved and printed as a title on the output. These five cards may be left blank, but must be included ahead of the first NAMELIST deck in the data. This descriptive message is read only once at the beginning of the program exectuion. The multiple case feature of running successive computations is accomplished by supplying multiple NAMELIST data sets with the changed variables indicated.

The following is a description of the input variables required for execution of the program. Where appropriate, default or suggested values are indicated. The following data are supplied through NAMELIST "DATA":

Program Option Parameters

IOPT:	Ramberg-Osgood parameter sentinel. IOPT=1: Input R-O parameter directly. IOPT=2: Determine parameters from stress-strain curve-fit routine.
COPT:	Curve-fit sentinel, (exercised if IOPT=2) COPT=1: Same stress-strain data for each layer. COPT=2: Different sets of data for each layer.

EOPT:	Exponent option, (exercised if IOPT=2) EOPT=1: Determine exponents from curve-fit routine. EOPT=2: Input exponents to curve-fit routine.

Solution Accuracy Parameters

KSGM:	No. of load incrememts, maximum = 50 increments.
SMLT:	Load increment multiple, maximum number of Newton-Raphson iterations, default is 100.
EPS:	Convergence criteria for Newton-Raphson analysis, default value is 10^{-3}.
UPBD:	Divergence criteria during Newton-Raphson analysis, default value is 20000.
INMT:	Incrementation estimate method, default value is 2.

Layer Description

NLAY:	Number of laminate layers (max. is 20).
THICK(LAY):	Thickness of each layer.
THETA(LAY):	Orientation of each layer in degrees.
MATYPE(LAY):	Material kind of each layer (maximum number of different materials is 20).

Material Description

E11	(MATYPE):	Lamina longitudinal modulus.
E22	"	Lamina transverse modulus.
G12	"	Lamina shear modulus.
V12	"	Lamina major Poisson's ratio.
S11T	"	Lamina longitudinal tensile strength.
S11C	"	Lamina longitudinal compressive strength.
S22T	"	Lamina transverse tensile strength.

S22C (MATYPE):	Lamina transverse compressive strength.
S12 "	Lamina in-plane shear strength.
EP11T "	Lamina longitudinal tensile strain.
EP11C "	Lamina longitudinal compressive strain.
EP22T "	Lamina transverse tensile strain.
EP22C "	Lamina transverse compressive strain.
GAMA "	Lamina in-plane shear strain.
A12 "	Lamina interaction term for quadratic interaction criteria - default value is 0.0.
STY "	Ramberg-Osgood tension constant
SCY "	Ramberg-Osgood compression constant
TY "	Ramberg-Osgood shear constant.
XM "	Ramberg-Osgood shear exponent (default value is 3.0).
XN "	Ramberg-Osgood tension exponent (default value is 3.0).

Stress-Strain Data (input in IOPT=2)

IPTS:	Number of stress-strain data points.
SIG11(I,MATYPE):	IPTS values of longitudinal lamina stresses for each material type.
SIG22 "	IPTS values to transverse lamina stresses for each material type.
SIG12 "	IPTS values of in-plane shear stresses for each material type.
EPS11 "	IPTS values of longitudinal lamina strains for each material type.
EPS22 "	IPTS values of transverse lamina strains for each material type.
EPS12 "	IPTS values of in-plane shear strain for each material type.

Applied Loading and Failure Criteria

SO11:	Initial axial stress applied to laminate.
SO22:	Initial transverse stress applied to laminate.
SO12:	Initial shear stress applied to laminate.
IFCN:	Failure criteria sentinel IFCN=1: ultimate stress IFCN=2: ultimate strain IFCN=3: quadratic interaction IFCN=4: all failure criteria
STIFF:	Ratio of final to initial laminate stiffness which constitutes failure due to stiffness reduction, default value is 0.10.

3. UNI PROGRAM

3.1 Description

Program UNI computes the elastic properties, thermal expansion coefficients and the Ramberg-Osgood shear stress parameter for the unidirectional fiber bundle or the lamina. The fiber may be isotropic or transversely isotropic and the matrix is isotropic. The effective elastic properties of the composite are calculated from the composite cylinder assemblage model proposed by Hashin and Rosen [2] and the thermal expansion coefficients from the analytical results of Ref. [3]. The program also calculates the Ramberg-Osgood shear stress parameter of the matrix as outlined in Ref. [1].

The UNI program takes constituent properties as input and computes laminae properties as output. The output properties provide all the information required as input to the NOLIN program. The program accepts families of fiber and matrix materials such that an array of laminae properties can be generated. This feature can be especially useful for sensitivity analyses. All input to UNI is accomplished through a single NAMELIST statement "UNID".

3.2 Input Description

The following describes the variables required for the execution of UNI through NAMELIST "UNID".

Program Control Variables

NF:	No. of fibers, max. is 20.
NM:	No. of matrices, max. is 20.
NVM:	No. of matrix vol. fractions, max. is 20.

Matrix Properties, J=1, NM

EM(J):	Young's modulus for Jth matrix.
RHOM(J):	Density of Jth matrix.
ANUM(J):	Poisson ratio for Jth matrix.
ALPM(J):	Coef. of thermal expansion for Jth matrix required only if ISTS=1.
ROMS(J):	Shear stress Ramberg-Osgood parameter for Jth matrix required only if NONLIN = 1.

Isotropic Fiber Properties, J=1, NF

The following variables are required when ISOT=1:

EF(J):	Young's modulus for Jth fiber.
ANUF(J):	Poisson ratio for Jth fiber.
RHOF(J):	Density of Jth fiber.
ALPF(J):	Coef. of thermal expansion for Jth fiber (required only if ISTS=1).

Transversely Isotropic Fiber Properties, J=1 NF

The following variables are required when ISOT=2:

EFA(J):	Axial Young's modulus for Jth fiber.
EFT(J):	Transverse Young's modulus for Jth fiber.
ANUFA(J):	Axial Poisson ratio for Jth fiber.
GFA(J):	Axial shear modulus for Jth fiber.
ANUFT(J):	Transverse Poisson ratio for Jth fiber.
ALPF(J):	Axial thermal expan. coef. for Jth fiber (required only if ISTS=1).
ALPFT(J):	Transverse thermal exp. coef. for Jth fiber. (required only if ISTS=1).

Laminae Volume Fractions, J=1, NVM

VM(J):	Jth volume fraction of matrix material.

The UNI program computes laminae properties for all combinations of fibers, matrix materials, and volume fractions which are supplied as input. That is, there are a total of NF x NM x NVM materials formed from the input. Within a given run the total number of materials (NFxNMxNVM) must be less than 200.

The following is a description of the properties computed by UNI and printed as output:

Calculated Thermo-Elastic Constituent Parameters

GFT(J):	Transverse shear modulus for Jth fiber.
GF(J):	Shear modulus for Jth fiber.
GM(J):	Shear mdoulus for Jth matrix.
AKF(J):	Plane strain bulk modulus for Jth fiber.
AKM(J):	Plane strain bulk modulus for Jth matrix.

Effective Thermo-Elastic Parameters

AKTS(J):	Effective trans. bulk modulus for Jth material.
EAS(J):	Effective axial Young's modulus for Jth material.
ETS(J):	Effective trans. Young's modulus for Jth material.
ANUAS(J):	Eff. Poisson ratio (unidirectional axial stress) for Jth material.
ANUTS(J):	Eff. Poisson ratio (in transverse plane) for Jth material.
GAS(J):	Eff. shear modulus (in fiber planes) for Jth material.
GTS(J):	Eff. shear mdoulus (in trans. planes) for Jth material.

ALPAS(J):	Eff. (fiber direction) thermal exp. coef. for Jth material.
ALPTS(J):	Eff. (trans. direction) thermal exp. coef. for Jth material.
RHOS(J):	Bulk density for Jth material.
ROCOMP(J):	Ramberg-Osgood shear stress parameter for Jth material.

4. SAMPLE PROBLEMS

The purpose of this section is to present several sample problems which illustrate the capabilities of both the UNI and the NOLIN programs. Both the program input and output are listed at the end of this section to aid in understanding the program.

4.1 UNI - Sample Problems

4.1.1 Laminae properties for Thornel 50 fibers in two carbon matrices have been determined. The transversely isotropic fiber option has been used to model the Thornel 50 fibers, and a Ramberg-Osgood shear stress parameter is included for the two matrix materials.

The input data and the resulting output data are given in sections 4.1.2 and 4.1.3. Note that the constituent properties are mirrored in the program output along with the required computed constituent properties. The effective thermoelastic properties for the 1-D laminae are printed on the second page of output. Units are consistent throughout the program such that the units need only be consistent for the constituent properties, in this case properties were input as MN/M^2.

The second sample problem combines KEVLAR-49 fibers with a range of matrix properties to obtain the 1-D composite properties. A list of input data for this case and the corresponding program output are shown in sections 4.1.2 and 4.1.3. It is interesting to note that an order of magnitude change in matrix modulus results in a factor of five change in transverse modulus and transverse shear modulus.

4.1.2 UNI SAMPLE PROBLEMS - INPUT CARDS

```
C   UNI   DATA

 $DATA[illegible]
 NF=1,    NM=2,    NV[illegible],
 EFA(1)=3.[illegible]E+0[illegible],    [illegible]FT(1)=7.[illegible]+03,    GFA[illegible]1[illegible]= .[illegible]E+04,
 ANUFA(1)=0.[illegible],    ANUFT(1)=0.1,
 RHOF([illegible])=1.,
 EM(1)=1.7E+[illegible]4, [illegible],
 ANUM(1)=2*0.2,
 RHOM(1)=2*1.,
 VM(1)=0.4,0.6,
 ROMS(1)=2*3.0,
 ALPF(1)=5.E-07,
 ALPFT(1)=3.[illegible]E-0[illegible],
 ALPM(1)=2*1.4E-[illegible]
 $DATA[illegible]NE
 NF=1,    NM=10,    NV[illegible],
 EFA(1)=1.[illegible]E+0[illegible],    ANUFA(1)=[illegible].2,    RHOF(1[illegible]=[illegible].,
 EFT(1)=9.7[illegible]E+03,    ANUFT(1)=0.2,    GFA(1[illegible]=[illegible]7E+03,
 EM(1)=.5E+0[illegible], 1.[illegible], 1.5E+0[illegible], 2.E+03, [illegible]E+03, 3.E+03,
       3.5E+03, 4.[illegible], 4.5E+03, 5.E+0[illegible],
 ANUM(1)=10*0.2,    RHOM(1)=10*1.,
 VM(1)=0.4.
 ALPF(1)= [illegible]6.1[illegible]-[illegible]7,    ALPFT(1)=[illegible].83E-06,
 ALPM(1)=10*1.39[illegible]
```

4.1.3 UNI SAMPLE PROBLEM OUTPUT
NO. 1

FIBER NO.	E(F)	NU(F)	G(F)	K(F)	RHO(F)	ALPHA(F)
1	3.60000E+05	.10000	13800	4018.4	1.0000	5.00000E-07
1	7230.0	.10000	3286.4	4018.4	1.0000	3.60000E-06

MATRIX NO.	E(M)	NU(M)	G(M)	K(M)	RHO(M)	ALPHA(M)
1	17000	.20000	7083.3	11806	1.0000	1.40000E-06
2	34000	.20000	14167	23611	1.0000	1.40000E-06

MATRIX NO.	R-O-S(M)
1	3.0000
2	3.0000

EFFECTIVE THERMOELASTIC PARAMETERS

F	M	MATERIAL	V(M)	E(A)* / E(T)*	NU(A)* / NU(T)*	G(A)* / G(T)*	K(T)* / RHO*	ALPHA(A)* / ALPHA(T)*
1	1	1	.40000	2.34833E+05	.15351	16471	6210.6	5.23229E-07
				10393	.16120	4475.1	1.00000	2.51547E-06
				R-O PARAMETER =	3.11794E+00			
1	1	2	.60000	1.62230E+05	.17215	9175.0	7667.0	5.52901E-07
				12392	.18709	5211.3	1.0000	2.13356E-06
				R-O PARAMETER =	2.96561E+00			
1	2	3	.40000	2.41649E+05	.16535	13946	8778.4	5.46548E-07
				14493	.17125	6186.8	1.00000	2.27233E-06
				R-O PARAMETER =	3.11794E+00			
1	2	4	.60000	1.72441E+05	.18093	14019	12234	6.01723E-07
				19563	.19299	8199.3	1.0000	1.94690E-06
				R-O PARAMETER =	2.96561E+00			

PROBLEM NO. 2 OUTPUT

FIBER NO.	E(F)	NU(F)	G(F)	K(F)	RHO(F)	ALPHA(F)
1	1.30000E+05	.20000	4170.0	6158.8	1.0000	-6.10000E-07
1	9780.0	.20000	4075.0	6158.8	1.0000	1.83000E-06

MATRIX NO.	E(M)	NU(M)	G(M)	K(M)	RHO(M)	ALPHA(M)
1	500.00	.20000	208.33	347.22	1.0000	1.39000E-05
2	1000.0	.20000	416.67	694.44	1.0000	1.39000E-05
3	1500.0	.20000	625.00	1041.7	1.0000	1.39000E-05
4	2000.0	.20000	833.33	1388.9	1.0000	1.39000E-05
5	2500.0	.20000	1041.7	1736.1	1.0000	1.39000E-05
6	3000.0	.20000	1250.0	2083.3	1.0000	1.39000E-05
7	3500.0	.20000	1458.3	2430.6	1.0000	1.39000E-05
8	4000.0	.20000	1666.7	2777.8	1.0000	1.39000E-05
9	4500.0	.20000	1875.0	3125.0	1.0000	1.39000E-05
10	5000.0	.20000	2083.3	3472.2	1.0000	1.39000E-05

MATRIX NO.	P-O-S(M)
1	3.0000
2	3.0000
3	0.
4	0.
5	0.
6	0.
7	0.
8	0.
9	0.
10	0.

EFFECTIVE THERMO-ELASTIC PARAMETERS

F	M	MATERIAL	V(M)	E(A)* E(T)*	NU(A)* NU(T)*	G(A)* G(T)*	K(T)* RHO*	ALPHA(A)* ALPHA(T)*
1	1	1	.40000	78200 1596,5	,20000 ,21564	598,35 656,64	1019,8 1,00000	-5,72890E-07 6,33022E-06
				R-O PARAMETER =	3,11794E+00			
1	2	2	,40000	78400 2803,9	,20000 ,21802	748,63 1151,0	1799,4 1,00000	-5,35969E-07 6,42491E-06
				R-O PARAMETER =	3,11794E+00			
1	3	3	,40000	78600 3760,0	,20000 ,21920	903,44 1542,0	2419,6 1,00000	-4,99237E-07 6,51767E-06
				R-O PARAMETER =	0,			
1	4	4	,40000	78800 4543,4	,20000 ,21973	1020,2 1862,4	2928,7 1,00000	-4,62690E-07 6,60856E-06
				R-O PARAMETER =	0,			
1	5	5	,40000	79000 5202,6	,20000 ,21990	1116,8 2132,4	3357,3 1,00000	-4,26329E-07 6,69761E-06
				R-O PARAMETER =	0,			
1	6	6	,40000	79200 5769,5	,20000 ,21985	1201,4 2364,8	3725,5 1,00000	-3,90152E-07 6,78490E-06
				R-O PARAMETER =	0,			
1	7	7	,40000	79400 6265,5	,20000 ,21969	1278,2 2568,5	4047,5 1,00000	-3,54156E-07 6,87046E-06
				R-O PARAMETER =	0,			
1	8	8	,40000	79600 6706,1	,20000 ,21945	1349,8 2749,6	4333,1 1,00000	-3,18342E-07 6,95435E-06
				R-O PARAMETER =	0,			
1	9	9	,40000	79800 7102,3	,20000 ,21917	1417,6 2912,8	4589,8 1,00000	-2,82797E-07 7,03662E-06
				R-O PARAMETER =	0,			
1	10	10	,40000	80000 7462,7	,20000 ,21887	1482,7 3061,3	4822,9 1,00000	-2,47250E-07 7,11730E-06
				R-O PARAMETER =	0,			

++++++++++++++++++++++++ PROGRAM TERMINATED ++++ ++++++++++++++++++++

4.2 NOLIN Sample Problems

4.2.1 The sample problems for the NOLIN program have been chosen to exercise several options of the program. Graphite/Epoxy and Boron/Aluminum laminates have been modeled, employing material properties derived from the Air Force Composites Design Guide.

The initial problem is a $\pm 30°$ Boron/Aluminum laminate under uniaxial tensile loading. Ramberg-Osgood parameters were input, with exponents set equal to 3.0, and a maximum strain failure criteria was employed. The input data and the program output follow in sections 4.2.2 and 4.2.3 respectively. A graph of the axial stress-strain curve for the laminate is given in Figure 2. Note that the laminate exhibits a non-linear stress-strain behavior from loading onset to failure.

The second sample nonlinear laminate analysis problem is a 0, ± 45 Graphite/Epoxy laminate under combined tension and shear loading. In this case the Ramberg-Osgood parameters were determined by the curve-fit routine from uniaxial stress-strain data supplied as input. The input data and program output listing are given in sections 4.2.2 and 4.2.3 respectively. Output of the program is plotted in the form of axial stress versus axial strain for this case in Figure 3.

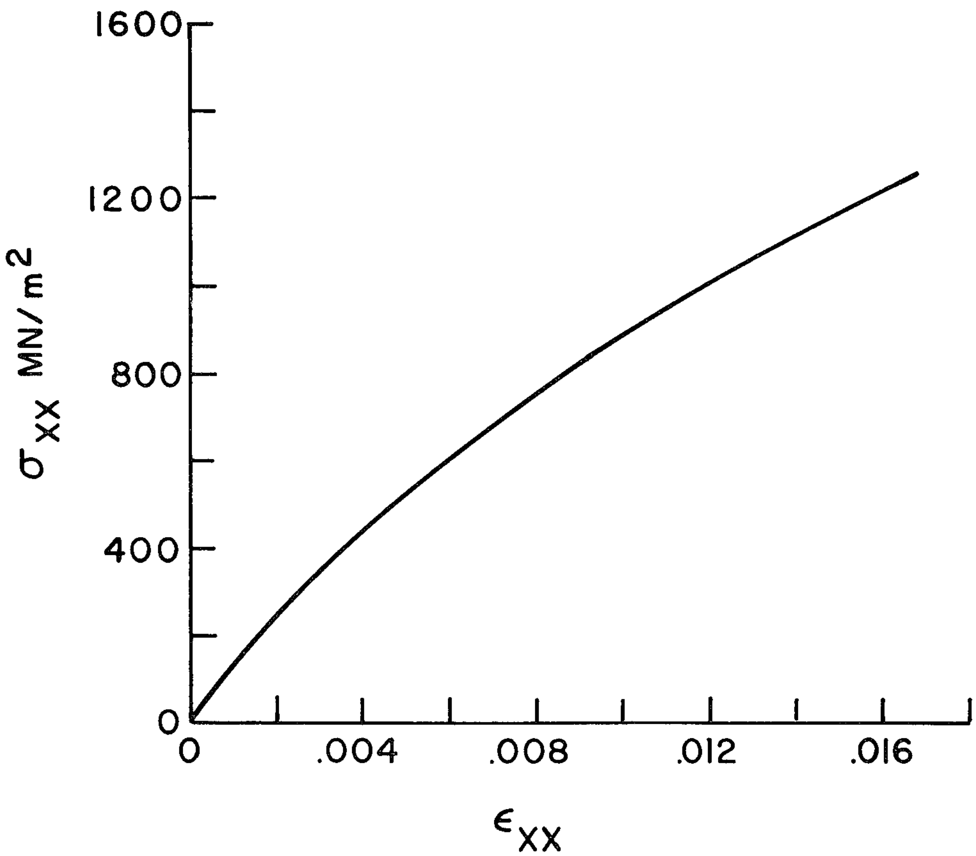

Figure 2 - $\pm 30°$ Boron/Aluminum Laminate - Axial Stress-Strain Behavior.

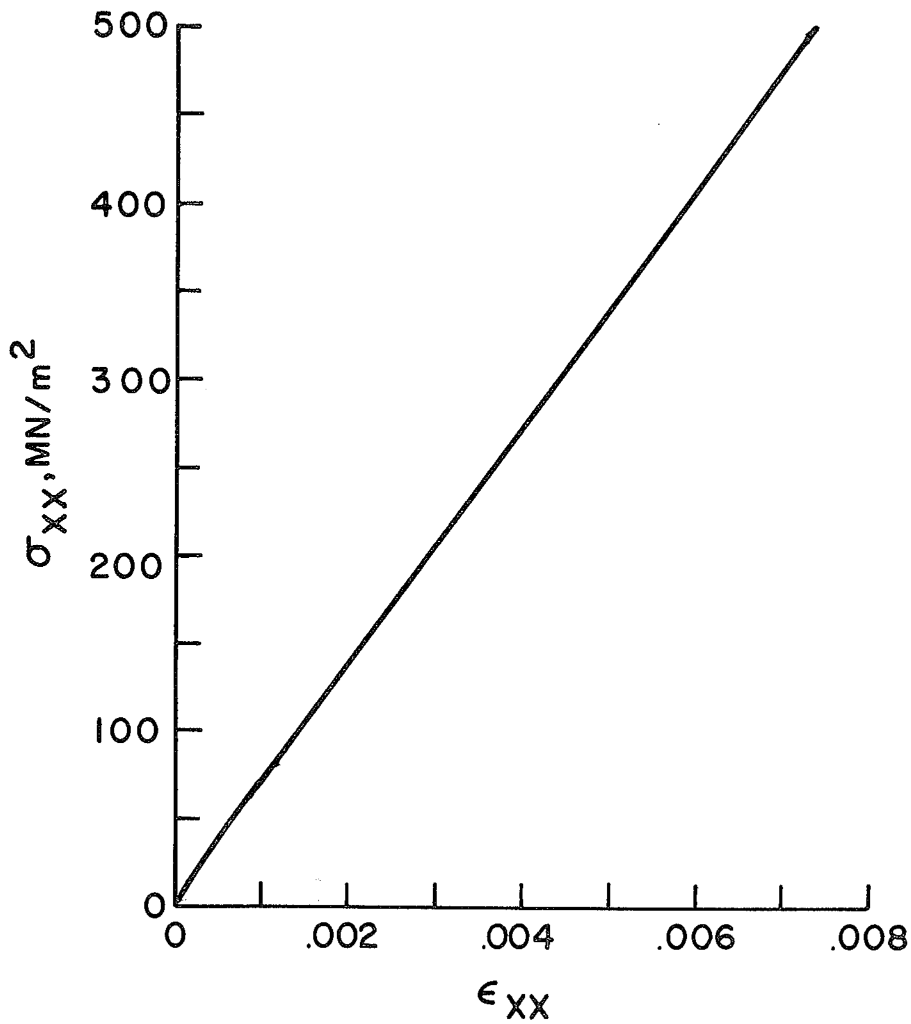

Figure 3 - Axial Stress-Strain Response of [0+45] Graphite-Epoxy Laminate under Axial and Shear Loading.

4.2.2 NOLIN SAMPLE PROBLEM INPUT CARDS

```
C  NOLIN DATA
```

```
*
*    NOLIN SAMPLE PROBLEMS                                 VERS 2 MOD 3
*
*    1.   0,+45,-45 LAMINATE, FULL R-O PARAM CURVE-FIT, ALL FAIL CRIT,
*    2.   +30,-30 LAMINATE, R-O PARAM INPUT, STRAIN FAIL CRIT,
 $DATA
 NLAY=3,
 E11(1)=3*1.982E+05,
 E22(1)=3*1.900E+04,
 V12(1)=3*0.255E+00,
 G12(1)=3*5.770E+03,
 THICK(1)=0.25,0.5,0.25,
 THETA(1)=45.,0.,-45.,
 IOPT=2,EOPT=1,COPT=1,
 IPTS=10,
 SIG11(1,1)=6.89E+00,13.78E+00,20.67E+00,27.56E+00,34.45E+00,41.34E+00,
        48.23E+00,55.12E+00,62.01E+00,68.90E+00,
 EPS11(1,1)=1.1E-03,2.6E-03,4.E-03,5.7E-03,7.6E-03,1.E-02,1.31E-02,
         1.65E-02,2.15E-02,2.84E-02,
 SIG22(1,1)=6.89E+00,13.78E+00,20.67E+00,27.56E+00,34.45E+00,41.34E+00,
        48.23E+00,55.12E+00,62.01E+00,68.90E+00,
 EPS22(1,1)=1.1E-03,2.6E-03,4.E-03,5.7E-03,7.6E-03,1.E-02,1.31E-02,
         1.65E-02,2.15E-02,2.84E-02,
 SIG12(1,1)=6.89E+00,13.78E+00,20.67E+00,27.56E+00,34.45E+00,41.34E+00,
```

NOLIN SAMPLE INPUT CARDS CONT.

```
      48.23E+00,55.12E+00,62.01E+00,68.90E+00,
EPS12(1,1)=2.2E-03,5.2E-03,8.E-03,11.4E-03,15.2E-03,2.E-02,2.62E-02,
3.30E-02,4.30E-02,5.68E-02,
SO11=+5.0E+00,SO22=-2.5E+00,SO12=0.0,
IFCN=4,IPRINT=1,
MATYPE(1)=1,1,1,
S11T(1)=1.32E03,S22T(1)=7.16E+01,S12(1)=1.05E+02,EP11T(1)=6.68E-03,
EP22T(1)=3.77E-03,GAMA(1)=1.827E-02,S11C(1)=2.43E+03,S22C(1)=2.75E+02,
EP11C(1)=1.227E-02,EP22C(1)=1.45E-02,
STIFF=0.100,
A12(1)= 3*-2.8623E-06,
KSGM=50,SMLT=3.0,IT=10,INMT=2$
$DATA
NLAY=2,
E11(1)=2*2.20E+05,
E22(1)=2*1.24E+05,
V12(1)=2*0.01E+00,
G12(1)=2*2.60E+04,
THICK(1)=2*0.50,
THETA(1)=30.0,-30.0,
IOPT=1,
STY(1)=2*1109.E+00,
SCY(1)=2*1467.E+00,
TY(1)=2*93.0E+00,
XM=3.00,XN=3.00,
SO11=50.,
SO22=0.,
SO12=0.,
IFCN=2,
S11T(1)=1100.0,S22T(1)=103.0,S12(1)=93.0,EP11T(1)=0.7E-02,
EP22T(1)=2.0E-02,GAMA(1)=3.0E-02,S11C(1)=1480.0,S22C(1)=160.,
EP11C(1)=0.7E-02,EP22C(1)=0.02,
STIFF=0.100,
KSGM=48,SMLT=4$
```

4.2.3 NOLIN SAMPLE OUTPUT

```
++++++++++++++++++++++++++++++++++++++++
+                                      +
+                                      +
+              NONLINEAR               +
+                                      +
+      THERMOELASTIC ANALYSIS          +
+                                      +
+                 OF                   +
+                                      +
+        FIBROUS COMPOSITES            +
+                                      +
+                AND                   +
+                                      +
+     NON-HOMOGENEOUS LAMINATES        +
+                                      +
+                                      +
+                                      +
++++++++++++++++++++++++++++++++++++++++

.VERSION 2     MOD 3  (MAY 74)

.DATE

.PROGRAM IDENTIFICATION

*
*    NOLIN SAMPLE PROBLEMS                        VERS 2 MOD 3
*
*    1.  0,+45,-45 LAMINATE. FULL R-O PARAM CURVE-FIT. ALL FAIL CRIT.
*    2.  +30,-30 LAMINATE. R-O PARAM INPUT. STRAIN FAIL CRIT.
```

LAMINATE 1
****

NUMBER OF LAYERS = 3

DATA INPUT POINTS FOR CURVE FIT-

```
6.89000E+00   1.37800E+01   2.06700E+01   2.75600E+01   3.44500E+01   4.13400E+01   4.82300E+01   5.51200E+01
6.20100E+01   6.89000E+01

1.10000E-03   2.60000E-03   4.00000E-03   5.70000E-03   7.60000E-03   1.00000E-02   1.31000E-02   1.65000E-02
2.15000E-02   2.84000E-02

6.89000E+00   1.37800E+01   2.06700E+01   2.75600E+01   3.44500E+01   4.13400E+01   4.82300E+01   5.51200E+01
6.20100E+01   6.89000E+01

1.10000E-03   2.60000E-03   4.00000E-03   5.70000E-03   7.60000E-03   1.00000E-02   1.31000E-02   1.65000E-02
2.15000E-02   2.84000E-02

6.89000E+00   1.37800E+01   2.06700E+01   2.75600E+01   3.44500E+01   4.13400E+01   4.82300E+01   5.51200E+01
6.20100E+01   6.89000E+01

2.20000E-03   5.20000E-03   8.00000E-03   1.14000E-02   1.52000E-02   2.00000E-02   2.62000E-02   3.30000E-02
4.30000E-02   5.68000E-02
```

LAYER	THETA	T	E11	E22	V12	V21	G12	SGT Y	SGC Y	TAUY
1	45.00	2.5000E-01	1.9820E+05	1.9000E+04	2.5500E-01	2.4445E-02	5.7700E+03	2.1069E+00	2.1069E+00	6.5456E+01
2	0.00	5.0000E-01	1.9820E+05	1.9000E+04	2.5500E-01	2.4445E-02	5.7700E+03	2.1069E+00	2.1069E+00	6.5456E+01
3	-45.00	2.5000E-01	1.9820E+05	1.9000E+04	2.5500E-01	2.4445E-02	5.7700E+03	2.1069E+00	2.1069E+00	6.5456E+01

EQUATION PARAMETERS

EXPONENT M = 2.74596E+00
EXPONENT N = 1.47386E+00

EXTERNALLY APPLIED STRESS
****

	INITIAL STRESS	STRESS INCREMENT	NO. OF INCREMENTS
SG XX	5.00000E+00	1.50000E+01	50
SG YY	-2.50000E+00	-7.50000E+00	
SG XY	0.	0.	

LAMINA FAILURE CRITERIA

```
                    ++++++++++++++++++++++
                    ALL FAILURE CRITERIA

LAYER               LL                    TT                    LT

          ULT. STRESS
          ULT. STRAIN

          NOTE:  ALL STRAINS ARE ENGINEERING COMPONENTS

  1    TENS.     1.32000E+03          7.16000E+01          1.05000E+02
       COMP.     2.43000E+03          2.75000E+02          1.05000E+02

  1    TENS.     6.68000E-03          3.77000E-03          1.82700E-02
       COMP.     1.22700E-02          1.45000E-02          1.82700E-02

  2    TENS.     1.32000E+03          7.16000E+01          1.05000E+02
       COMP.     2.43000E+03          2.75000E+02          1.05000E+02

  2    TENS.     6.68000E-03          3.77000E-03          1.82700E-02
       COMP.     1.22700E-02          1.45000E-02          1.82700E-02

  3    TENS.     1.32000E+03          7.16000E+01          1.05000E+02
       COMP.     2.43000E+03          2.75000E+02          1.05000E+02

  3    TENS.     6.68000E-03          3.77000E-03          1.82700E-02
       COMP.     1.22700E-02          1.45000E-02          1.82700E-02

LAYER               QUADRATIC INTERACTION TERM (A12)

  1                          -2.86230E-06
  2                          -2.86230E-06
  3                          -2.86230E-06

STIFFNESS =      1.00000E-01

                    CONTROL PARAMETERS
                    ++++++++++++++++++

MAX. NO. OF ITERATIONS =     10
CONVERGENCE CRITERIA   =      1.00000E-03
DIVERGENCE CRITERIA    =      2.00000E+04

                 LAMINATE CONSTANTS (STRESS-STRAIN)
                 ++++++++++++++++++++++++++++++++++++

                    EXX  =      1.11891E+05
                    EYY  =      3.49664E+04
                    VYX  =      6.85439E-01
                    VXY  =      2.14203E-01
                    GXY  =      2.89865E+04
```

APPLIED STRESS ANALYSIS

EXTERNAL APPLIED STRESS

SG XX = 5.00000F+00
SG YY = -2.50000E+00
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 2 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-5.44278E+00	-4.36713E-01	-9.62175E-01	6.18445E-05	-1.15643E-04	-7.23315E-13	-2.68992E-05	-2.68992E-05	-8.87437E-05
2	1.19776E+01	-1.09807E+00	0.	6.18445E-05	-1.15643E-04	0.	6.18445E-05	-1.15643E-04	0.
3	-5.44278E+00	-4.36713E-01	9.62182E-01	6.18452E-05	-1.15644E-04	-7.41279E-13	-2.68992E-05	-2.68992E-05	8.87444E-05

EXTERNAL APPLIED STRESS

SG XX = 2.00000E+01
SG YY = -1.00000E+01
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 3 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-2.54934E+01	-1.60973E+00	-2.74989E+00	2.61075E-04	-5.14182E-04	2.48289E-11	-1.26554E-04	-1.26554E-04	-3.87628E-04
2	5.08018E+01	-3.69865E+00	0.	2.61075E-04	-5.14182E-04	0.	2.61075E-04	-5.14182E-04	0.
3	-2.54934E+01	-1.60973E+00	2.74964E+00	2.61039E-04	-5.14147E-04	2.45162E-11	-1.26554E-04	-1.26554E-04	3.87593E-04

EXTERNAL APPLIED STRESS

SG XX = 3.50000E+01
SG YY = -1.75000E+01
SG XY = 0.

SOLUTION FOR STRFSS CONVERGES WITHIN 4 ITERATIONS

	STRFSS (LAYFR AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-4.87608E+01	-2.72327E+00	-3.22333E+00	4.74559E-04	-9.59587E-04	-2.64055E-12	-2.42514E-04	-2.42514E-04	-7.17073E-04
2	9.25187E+01	-6.03463E+00	0.	4.74559E-04	-9.59587E-04	0.	4.74559E-04	-9.59587E-04	0.
3	-4.87608E+01	-2.72327E+00	3.22336E+00	4.74565E-04	-9.59594E-04	-2.64345E-12	-2.42514E-04	-2.42514E-04	7.17080E-04

EXTERNAL APPLIED STRESS

SG XX = 5.00000E+01

```
SG YY  =  -2.50000E+01
SG XY  =   0.

SOLUTION  FOR  STRESS  CONVERGES  WITHIN   3  ITERATIONS

                     STRESS                                  STRAIN                                  STRAIN
                  (LAYER AXES)                          (LAMINATE AXES)                           (LAYER AXES)
LAYER    SG 11         SG 22         SG 12        EP XX         EP YY         EP XY          EP 11         EP 22         EP 12

  1  -7.33712E+01 -3.76905E+00 -3.24695E+00   6.93289E-04 -1.42397E-03 -3.78710E-12   -3.65339E-04 -3.65339E-04 -1.05863E-03
  2   1.45323E+02 -8.18288E+00  0.            6.93289E-04 -1.42397E-03  0.             6.93289E-04 -1.42397E-03  0.
  3  -7.33712E+01 -3.76905E+00  3.24699E+00   6.93302E-04 -1.42398E-03 -3.79320E-12   -3.65339E-04 -3.65339E-04  1.05864E-03

EXTERNAL APPLIED STRESS

SG XX  =   6.50000E+01
SG YY  =  -3.25000E+01
SG XY  =   0.

SOLUTION  FOR  STRESS  CONVERGES  WITHIN   2  ITERATIONS

                     STRESS                                  STRAIN                                  STRAIN
                  (LAYER AXES)                          (LAMINATE AXES)                           (LAYER AXES)
LAYER    SG 11         SG 22         SG 12        EP XX         EP YY         EP XY          EP 11         EP 22         EP 12

  1  -9.85741E+01 -4.75320E+00 -3.15385E+00   9.13755E-04 -1.89622E-03 -2.15423E-12   -4.91231E-04 -4.91231E-04 -1.40499E-03
  2   1.78510E+02 -1.01825E+01  0.            9.13755E-04 -1.89622E-03  0.             9.13755E-04 -1.89622E-03  0.
  3  -9.85741E+01 -4.75320E+00  3.15387E+00   9.13765E-04 -1.89623E-03 -2.15636E-12   -4.91231E-04 -4.91231E-04  1.40500E-03

EXTERNAL APPLIED STRESS

SG XX  =   8.00000E+01
SG YY  =  -4.00000E+01
SG XY  =   0.

SOLUTION  FOR  STRESS  CONVERGES  WITHIN   2  ITERATIONS

                     STRESS                                  STRAIN                                  STRAIN
                  (LAYER AXES)                          (LAMINATE AXES)                           (LAYER AXES)
LAYER    SG 11         SG 22         SG 12        EP XX         EP YY         EP XY          EP 11         EP 22         EP 12

  1  -1.24105E+02 -5.68706E+00 -3.03740E+00   1.13489E-03 -2.37258E-03  3.92971E-12   -6.18844E-04 -6.18844E-04 -1.75374E-03
  2   2.21859E+02 -1.20665E+01  0.            1.13489E-03 -2.37258E-03  0.             1.13489E-03 -2.37258E-03  0.
  3  -1.24105E+02 -5.68706E+00  3.03736E+00   1.13487E-03 -2.37256E-03  3.92203E-12   -6.18844E-04 -6.18844E-04  1.75372E-03

EXTERNAL APPLIED STRESS

SG XX  =   9.50000E+01
SG YY  =  -4.75000E+01
SG XY  =   0.

SOLUTION  FOR  STRESS  CONVERGES  WITHIN   2  ITERATIONS
  :
ETC.
  :
```

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-7.09459E+02	-2.12599E+01	-1.91901E+00	6.02590E-03	-1.31302E-02	2.76845E-12	-3.55216E-03	-3.55216E-03	-9.57806E-03
2	1.18344E+03	-4.27218E+01	0.	6.02590E-03	-1.31302E-02	0.	6.02590E-03	-1.31302E-02	0.
3	-7.09459E+02	-2.12599E+01	1.91900E+00	6.02585E-03	-1.31302E-02	2.75785E-12	-3.55216E-03	-3.55216E-03	9.57801E-03

EXTERNAL APPLIED STRESS

SG XX = 4.25000E+02
SG YY = -2.12500E+02
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-7.36579E+02	-2.18573E+01	-1.89844E+00	6.24879E-03	-1.36252E-02	2.50081E-12	-3.68822E-03	-3.68822E-03	-9.93701E-03
2	1.22732E+03	-4.38833E+01	0.	6.24879E-03	-1.36252E-02	0.	6.24879E-03	-1.36252E-02	0.
3	-7.36579E+02	-2.18573E+01	1.89843E+00	6.24874E-03	-1.36252E-02	2.49188E-12	-3.68822E-03	-3.68822E-03	9.93696E-03

EXTERNAL APPLIED STRESS

SG XX = 4.40000E+02
SG YY = -2.20000E+02
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-7.63727E+02	-2.24492E+01	-1.87882E+00	6.47171E-03	-1.41206E-02	2.26652E-12	-3.82443E-03	-3.82443E-03	-1.02961E-02
2	1.27121E+03	-4.50331E+01	0.	6.47171E-03	-1.41206E-02	0.	6.47171E-03	-1.41206E-02	0.
3	-7.63727E+02	-2.24492E+01	1.87881E+00	6.47166E-03	-1.41205E-02	2.25905E-12	-3.82443E-03	-3.82443E-03	1.02961E-02

LAMINATE HAS FAILED, EP 11 EXCEEDS MAXIMUM

AT FIRST POST-FAILURE LOAD POINT
EXTERNAL APPLIED STRESS

SG XX = 4.55000E+02
SG YY = -2.27500E+02
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

STRESS STRAIN STRAIN

	(LAYER AXES)			(LAMINATE AXES)			(LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-7.40900E+02	-2.34356E+01	-1.86008E+00	6.69466E-03	-1.46162E-02	2.06077E-12	-3.96078E-03	-3.96078E-03	-1.06554E-02
2	1.31511E+03	-4.61719E+01	0.	6.69466E-03	-1.46162E-02	0.	6.69466E-03	-1.46162E-02	0.
3	-7.40900E+02	-2.34356E+01	1.86007E+00	6.69462E-03	-1.46162E-02	2.05438E-12	-3.96078E-03	-3.96078E-03	1.06554E-02

**** LAMINATE ANALYSIS INTERPOLATED TO FAILURE POINT ****

AT FAILURE
EXTERNAL APPLIED STRESS

SG XX = 4.54014E+02
SG YY = -2.27007E+02
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 2 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-7.89113E+02	-2.29972E+01	-1.86128E+00	6.68000E-03	-1.45836E-02	1.07969E-13	-3.95181E-03	-3.95181E-03	-1.06318E-02
2	1.31222E+03	-4.69974E+01	0.	6.68000E-03	-1.45836E-02	0.	6.68000E-03	-1.45836E-02	0.
3	-7.89113E+02	-2.29972E+01	1.86128E+00	6.68000E-03	-1.45836E-02	1.07941E-13	-3.95181E-03	-3.95181E-03	1.06318E-02

LAMINATE HAS FAILED, SG 11 EXCEEDS MAXIMUM

AT FIRST POST-FAILURE LOAD POINT
EXTERNAL APPLIED STRESS

SG XX = 4.70000E+02
SG YY = -2.35000E+02
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-8.18099E+02	-2.36168E+01	-1.84215E+00	6.91764E-03	-1.51122E-02	1.87914E-12	-4.09726E-03	-4.09726E-03	-1.10149E-02
2	1.35902E+03	-4.73000E+01	0.	6.91764E-03	-1.51122E-02	0.	6.91764E-03	-1.51122E-02	0.
3	-8.18099E+02	-2.36168E+01	1.84215E+00	6.91761E-03	-1.51121E-02	1.87378E-12	-4.09726E-03	-4.09726E-03	1.10149E-02

**** LAMINATE ANALYSIS INTERPOLATED TO FAILURE POINT ****

AT FAILURE
EXTERNAL APPLIED STRESS

SG XX = 4.56671E+02
SG YY = -2.28336E+02
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 2 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-7.93930E+02	-2.31006E+01	-1.85804E+00	6.71950E-03	-1.46715E-02	1.96310E-12	-3.97598E-03	-3.97598E-03	-1.06955E-02
2	1.32000E+03	-4.62981E+01	0.	6.71950E-03	-1.46715E-02	0.	6.71950E-03	-1.46715E-02	0.
3	-7.93930E+02	-2.31006E+01	1.85803E+00	6.71947E-03	-1.46714E-02	1.95735E-12	-3.97598E-03	-3.97598E-03	1.06954E-02

EXTERNAL APPLIED STRESS

SG XX = 4.85000E+02
SG YY = -2.42500E+02
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-8.45321E+02	-2.41931E+01	-1.82498E+00	7.14066E-03	-1.56084E-02	1.71832E-12	-4.23386E-03	-4.23386E-03	-1.13745E-02
2	1.40293E+03	-4.84179E+01	0.	7.14066E-03	-1.56084E-02	0.	7.14066E-03	-1.56084E-02	0.
3	-8.45321E+02	-2.41931E+01	1.82498E+00	7.14062E-03	-1.56084E-02	1.71377E-12	-4.23386E-03	-4.23386E-03	1.13745E-02

EXTERNAL APPLIED STRESS

SG XX = 5.00000E+02
SG YY = -2.50000E+02
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-8.72567E+02	-2.47645E+01	-1.80852E+00	7.36370E-03	-1.61049E-02	1.57546E-12	-4.37059E-03	-4.37059E-03	-1.17343E-02
2	1.44686E+03	-4.95259E+01	0.	7.36370E-03	-1.61049E-02	0.	7.36370E-03	-1.61049E-02	0.
3	-8.72567E+02	-2.47645E+01	1.80851E+00	7.36367E-03	-1.61049E-02	1.57158E-12	-4.37059E-03	-4.37059E-03	1.17343E-02

```
LAMINATE  HAS  FAILED+   QUADRATIC INTERACTION FAILURE
                          QUADRATIC =  1.0320 FOR LAYER  2
                          QUADRATIC =   .9714 FOR LAYER  2 OF PREVIOUS LOAD

AT  FIRST  POST-FAILURE  LOAD  POINT
EXTERNAL APPLIED STRESS

SG XX  =    5.15000E+02
SG YY  =   -2.57500E+02
SG XY  =    0.

SOLUTION  FOR  STRESS  CONVERGES  WITHIN   1  ITERATIONS
```

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-8.99834E+02	-2.53313E+01	-1.79271E+00	7.58678E-03	-1.66017E-02	1.44798E-12	-4.50744E-03	-4.50744E-03	-1.20942E-02
2	1.49079E+03	-5.06245E+01	0.	7.58678E-03	-1.66017E-02	0.	7.58678E-03	-1.66017E-02	0.
3	-8.99834E+02	-2.53313E+01	1.79270E+00	7.58675E-03	-1.66016E-02	1.44462E-12	-4.50744E-03	-4.50744E-03	1.20942E-02

```
**** LAMINATE ANALYSIS INTERPOLATED TO FAILURE POINT ****

AT  FAILURE
EXTERNAL APPLIED STRESS

SG XX  =    5.07082E+02
SG YY  =   -2.53541E+02
SG XY  =    0.

SOLUTION  FOR  STRESS  CONVERGES  WITHIN   2  ITERATIONS
```

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	-8.85438E+02	-2.50327E+01	-1.80097E+00	7.46902E-03	-1.63394E-02	7.18620E-13	-4.43519E-03	-4.43519E-03	-1.19042E-02
2	1.46760E+03	-5.00457E+01	0.	7.46902E-03	-1.63394E-02	0.	7.46902E-03	-1.63394E-02	0.
3	-8.85438E+02	-2.50327E+01	1.80097E+00	7.46900E-03	-1.63394E-02	7.17842E-13	-4.43519E-03	-4.43519E-03	1.19042E-02

LAMINATE 2
++++++++++++

NUMBER OF LAYERS = 2

LAYER	THETA	T	E11	E22	V12	V21	G12	SGT Y	SGC Y	TAUY
1	30.00	5.0000E-01	2.2000E+05	1.2400E+05	1.0000E-02	5.6364E-03	2.6000E+04	1.1090E+03	1.4670E+03	9.3000E+01
2	-30.00	5.0000E-01	2.2000E+05	1.2400E+05	1.0000E-02	5.6364E-03	2.6000E+04	1.1090E+03	1.4670E+03	9.3000E+01

EQUATION PARAMETERS

EXPONENT M = 3.00000E+00
EXPONENT N = 3.00000E+00

EXTERNALLY APPLIED STRESS
++++++++++++++++++++++++++

	INITIAL STRESS	STRESS INCREMENT	NO. OF INCREMENTS
SG XX	5.00000E+01	2.00000E+02	48
SG YY	0.	0.	
SG XY	0.	0.	

LAMINA FAILURE CRITERIA
++++++++++++++++++++++
ULTIMATE STRESS

ULT. STRAIN

NOTE: ALL STRAINS ARE ENGINEERING COMPONENTS

LAYER		LL	TT	LT
1	TENS.	7.00000E-03	2.00000E-02	3.00000E-02
	COMP.	7.00000E-03	2.00000E-02	3.00000E-02
2	TENS.	7.00000E-03	2.00000E-02	3.00000E-02
	COMP.	7.00000E-03	2.00000E-02	3.00000E-02

STIFFNESS = 1.00000E-01

CONTROL PARAMETERS
+++++++++++++++++

MAX. NO. OF ITERATIONS = 10
CONVERGENCE CRITERIA = 1.00000E-03
DIVERGENCE CRITERIA = 2.00000E+04

LAMINATE CONSTANTS (STRESS-STRAIN)
++++++++++++++++++++++++++++++++++

EXX = 1.31218E+05
EYY = 8.96343E+04

```
VYX  =    4.42435E-01
VXY  =    3.02224E-01
GXY  =    7.05386E+04
```

APPLIED STRESS ANALYSIS

EXTERNAL APPLIED STRESS

SG XX = 5.00000E+01
SG YY = 0.
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	5.37302E+01	-3.73016E+00	-1.22801E+01	3.83121E-04	-1.71772E-04	-2.69684E-13	2.44398E-04	-3.30489E-05	-2.40276E-04
2	5.37302E+01	-3.73016E+00	1.22801E+01	3.83121E-04	-1.71772E-04	-1.88746E-13	2.44398E-04	-3.30489E-05	2.40276E-04

EXTERNAL APPLIED STRESS

SG XX = 2.50000E+02
SG YY = 0.
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 2 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	2.80731E+02	-3.07312E+01	-5.44261E+01	2.08896E-03	-1.15710E-03	-8.72517E-10	1.27745E-03	-3.45581E-04	-1.40559E-03
2	2.80731E+02	-3.07313E+01	5.44261E+01	2.08896E-03	-1.15710E-03	-6.95787E-10	1.27745E-03	-3.45583E-04	1.40558E-03

EXTERNAL APPLIED STRESS

SG XX = 4.50000E+02
SG YY = 0.
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	5.25892E+02	-7.58573E+01	-8.61013E+01	4.17189E-03	-2.93967E-03	-3.02218E-07	2.39387E-03	-1.16165E-03	-3.07947E-03
2	5.25854E+02	-7.58890E+01	8.60956E+01	4.17132E-03	-2.93969E-03	-2.85248E-07	2.39370E-03	-1.16206E-03	3.07909E-03

EXTERNAL APPLIED STRESS

SG XX = 6.50000E+02
SG YY = 0.
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 2 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	7.80039E+02	-1.30051E+02	-1.12556E+02	6.64157E-03	-5.71879E-03	1.41357E-07	3.55154E-03	-2.62876E-03	-5.35216E-03
2	7.80052E+02	-1.30040E+02	1.12558E+02	6.64179E-03	-5.71877E-03	1.19534E-07	3.55160E-03	-2.62858E-03	5.35231E-03

EXTERNAL APPLIED STRESS

SG XX = 8.50000E+02
SG YY = 0.
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	1.03875E+03	-1.88653E+02	-1.36434E+02	9.53059E-03	-9.66849E-03	-1.48380E-06	4.73018E-03	-4.86807E-03	-8.31382E-03
2	1.03865E+03	-1.88747E+02	1.36422E+02	9.52838E-03	-9.66863E-03	-1.31694E-06	4.72970E-03	-4.86995E-03	8.31222E-03

EXTERNAL APPLIED STRESS

SG XX = 1.05000E+03
SG YY = 0.
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

	STRESS (LAYER AXES)			STRAIN (LAMINATE AXES)			STRAIN (LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	1.29984E+03	-2.49716E+02	-1.58908E+02	1.28870E-02	-1.49782E-02	-2.27314E-06	5.91972E-03	-8.01090E-03	-1.20665E-02
2	1.29971E+03	-2.49833E+02	1.58895E+02	1.28838E-02	-1.49784E-02	-1.99637E-06	5.91912E-03	-8.01372E-03	1.20642E-02

LAMINATE HAS FAILED. EP 11 EXCEEDS MAXIMUM

AT FIRST POST-FAILURE LOAD POINT
EXTERNAL APPLIED STRESS

SG XX = 1.25000E+03
SG YY = 0.
SG XY = 0.

SOLUTION FOR STRESS CONVERGES WITHIN 1 ITERATIONS

STRESS STRAIN STRAIN

	(LAYER AXES)			(LAMINATE AXES)			(LAYER AXES)		
LAYER	SG 11	SG 22	SG 12	EP XX	EP YY	EP XY	EP 11	EP 22	EP 12
1	1.56228E+03	-3.12157E+02	-1.80593E+02	1.67730E-02	-2.18526E-02	-2.67009E-06	7.11545E-03	-1.21950E-02	-1.67260E-02
2	1.56215E+03	-3.12271E+02	1.80581E+02	1.67695E-02	-2.18529E-02	-2.32925E-06	7.11488E-03	-1.21983E-02	1.67234E-02

++++++++++++++++++++++++ PROGRAM TERMINATED ++++++++++++++++++++++++

5. Computer Program Listings

Source listings of both the UNI and NOLIN computer programs follow. The UNI program requires 20 K of computer core storage in a CDC 6600 machine, while the NOLIN program requires 60 K of core storage. No peripheral devices are required for either program for intermediate data storage.

5.1 UNI LISTING

RUN VERSION 2.3 --PSR LEVEL 363--   07/09/74

```
                PROGRAM MAIN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
          C
          C      UNI
          C     UNI COMPUTES UNIDIRECTIONAL FIBER BUNDLE PROPERTIES, EXPANSION
          C      COEFFICIENTS AND RAMBERG-OSGOOD SHEAR STRESS PARAMETER FOR THE
          C       UNIDIRECTIONAL FIBER BUNDLE
          C
          C     PRESENT VERSION INCLUDES:
          C      1 : TRANSVERSELY ISOTROPIC FIBER/ISOTROPIC MATRIX
          C      2 : AXIAL AND TRANSVERSE FIBER BUNDLE EXPANSION COEFFICIENTS
          C      3 : RAMBERG-OSGOOD SHEAR STRESS PARAMETER FOR THE UNIDIRECTIONAL
          C       COMPOSITE AS DERIVED FROM THE RAMBERG-OSGOOD SHEAR STRESS PARAMETER
          C        FOR THE MATRIX
          C
          C                                  * INPUT PARAMETERS *
          C
          C      NF        : NO. OF FIBERS IN COMPOSITE
          C      NM        : NO. OF MATRICES IN COMPOSITE
          C      NVM       : NO. OF MATRIX VOL. FRACTIONS
          C      EF(J)     : YOUNG#S MOD. FOR JTH FIBER
          C      EM(J)     : YOUNG#S MOD. FOR JTH MATRIX
          C      ANUF(J)   : POISSON RATIO FOR JTH FIBER
          C      ANUM(J)   : POISSON RATIO FOR JTH MATRIX
          C      RHOF(J)   : DENSITY OF JTH FIBER
          C      RHOM(J)   : DENSITY OF JTH MATRIX
          C      ALPM(J)   : COEF. OF THERMAL EXPANSION FOR JTH MATRIX
          C      VM(J)     : VOL. FRACTION FOR JTH MATRIX
          C      ROMS(J)   : SHEAR STRESS RAMBERG OSGOOD PARAMETER FOR JTH MATRIX
          C      ALPF(J)   : COEF. OF THERMAL EXPANSION FOR JTH FIBER
          C      ALPFT(J)  : TRANSVERSE THERMAL EXP. COEF. FOR JTH FIBER
          C      EFA(J)    : AXIAL YOUNG#S MOD. FOR JTH FIBER
          C      EFT(J)    : TRANSVERSE YOUNG#S MOD. FOR JTH FIBER
          C      ANUFA(J)  : AXIAL POISSON RATIO FOR JTH FIBER
          C      GFA(J)    : AXIAL SHEAR MODULUS FOR JTH FIBER
          C      ANUFT(J)  : TRANSVERSE POISSON RATION FOR JTH FIBER
          C
 000003          COMMON    /AREA01/EAS(200),ETS(200),ANUAS(200),GAS(200),GTS(200),
                1                    AKTS(200),ANUTS(200)
 000003          COMMON    /AREA02/L
 000003          COMMON    /AREA03/RHOS(200),ALPHA(3,200)
 000003          COMMON    /AREA06/EM(20),GM(20),VM(20),ANUM(20),ALPM(20),AKM(20)
 000003          COMMON    /AREA07/AKF(20),ALPF(20),ALPFT(20)
 000003          COMMON    /AREA10/RIGKM(20),RIGKF(20)
 000003          COMMON    /AREA11/MF(200),MV(200),MM(200)
 000003          COMMON    /AREA20/ROCOMP(200),ROMS(20)
 000003          DIMENSION          EF(20),ANUF(20),RHOF(20),RHOM(20),GF(20)
 000003          DIMENSION          SM(3,3),SF(3,3)
 000003          DIMENSION     EFA(20),EFT(20),ANUFA(20),ANUFT(20),GFA(20),GFT(20)
 000003          EQUIVALENCE        (EF(1),EFA(1)),(GF(1),GFA(1)),(ANUF(1),ANUFA(1))
          C
          C                                  * VARIABLE DICTIONARY *
          C
          C      CALCULATED THERMO-ELASTIC CONSTITUENT PARAMETERS
          C      GFT(J)    : TRANSVERSE SHEAR MODULUS FOR JTH FIBER
```

```
         C      GF(J)    : SHEAR MODULUS FOR JTH FIBER
         C      GM(J)    : SHEAR MODULUS FOR JTH MATRIX
         C      AKF(J)   : PLANE STRAIN BULK MODULUS FOR JTH FIBER
         C      AKM(J)   : PLANE STRAIN BULK MODULUS FOR JTH MATRIX
         C
         C      EFFECTIVE THERMOELASTIC PARAMETERS
         C      AKTS(J) : EFFECTIVE TRANS. BULK MOD. FOR JTH MATERIAL
         C      EAS(J)  : EFFECTIVE AXIAL YOUNG#S MOD. FOR JTH MATERIAL
         C      ETS(J)  : EFFECTIVE TRANS. YOUNG#S MOD. FOR JTH MATERIAL
         C      ANUAS(J): EFF. POISSON RATIO (UNIDIRECTIONAL AX. STRESS) FOR JTH MATERIAL
         C      ANUTS(J): EFF. POISSON RATIO (IN TRANSVERSE PLANE) FOR JTH MATERIAL
         C      GAS(J)  : EFF. SHEAR MOD. (IN FIBER PLANES) FOR JTH MATERIAL
         C      GTS(J)  : EFF. SHEAR MOD. (IN TRANS. PLANES) FOR JTH MATERIAL
         C      ALPAS(J): EFF. (FIBER DIRECTION) THERMAL EXP. COEF. FOR JTH MATERIAL
         C      ALPTS(J): EFF. (TRANS. DIRECTION) THERMAL EXP. COEF. FOR JTH MATERIAL
         C      RHOS(J) : BULK DENSITY FOR JTH MATERIAL
         C      ROCOMP(J): RAMBERG-OSGOOD SHEAR STRESS PARAMETER FOR JTH MATERIAL
         C
         C                                  * PROGRAMMING INFORMATION *
         C
         C      UNI MAY BE USED AS A SUBROUTINE
         C      SUBROUTINES REQUIRED    : MSUB01,MSUB02
         C
         C      **************************************************************************
         C
         C   INITIALIZE VARIABLES TO ZERO.
000003          NAMELIST/DATAONE/NF,NM,NVM,EF,ANUF,RHOF,EFA,ANUFA,GFA,EFT,ANUFT,
               1EM,ANUM,RHOM,VM,ROMS,ALPF,ALPFT,ALPM
000003          DATA       EFT/20*0.0/
000003          DATA       ROMS/20*0.0/
000003          DATA       ALPM/20*0.0/
000003      171 CONTINUE
000003          READ(5,DATAONE)
000006          IF(EOF,5)172,173
000011      173 CONTINUE
000011          IF(EFT(1))1012,1012,1014
000013     1012 CONTINUE
000013          DO 3 J=1,NF
000015          IF (J.NE.1) GO TO 4
000017          WRITE(6,202)
000022          IF(ALPM(1))5,5,6
000024        5 WRITE(6,203)
000030          GO TO 4
000031        6 WRITE(6,204)
000035        4 GF(J) = EF(J)/((1.0+ANUF(J))*2.0)
000042          AKF(J) = GF(J)/(1.0-2.0*ANUF(J))
000046          IF(ALPM(1))8,8,9
000050        8 WRITE(6,205)(J,EF(J),ANUF(J),GF(J),AKF(J),RHOF(J))
000070          GO TO 3
000071        9 WRITE(6,206)(J,EF(J),ANUF(J),GF(J),AKF(J),RHOF(J),ALPF(J))
000113        3 CONTINUE
000116          GO TO 1018
000116     1014 CONTINUE
000116          DO 503 J=1,NF
000120          IF (J.NE.1) GO TO 504
```

```
000122         IF(ALPM(1))505,505,506
000123     505 WRITE(6,203)
000127         GO TO 504
000130     506 WRITE(6,204)
000134     504 CONTINUE
000134         GFT(J) = EFT(J)/((1.0+ANUFT(J))*2.0)
000141         AKF(J)=EFA(J)*EFT(J)/(2.*EFA(J)*(1.-ANUFT(J))-4.*EFT(J)*
              1       ANUFA(J)**2)
000154         IF(ALPM(1))508,508,509
000155     508 WRITE(6,205)(J,EFA(J),ANUFA(J),GFA(J),AKF(J),RHOF(J))
000175         WRITE(6,205)(J,EFT(J),ANUFT(J),GFT(J),AKF(J),RHOF(J))
000215         GO TO 503
000216     509 WRITE(6,206) (J,EFA(J),ANUFA(J),GFA(J),AKF(J),RHOF(J),ALPF(J))
000240         WRITE(6,206) (J,EFT(J),ANUFT(J),GFT(J),AKF(J),RHOF(J),ALPFT(J))
000262     503 CONTINUE
000265    1018 CONTINUE
000265         DO 10 J = 1,NM                                                   00019900
000267         IF (J.NE.1) GO TO 11                                             00020000
000271         IF(ALPM(1))12,12,13
000272      12 WRITE(6,207)                                                     00020200
000276         GO TO 11                                                         00020300
000277      13 WRITE(6,208)                                                     00020400
000303      11 GM(J) = EM(J)/((1.0+ANUM(J))*2.0)                                00020500
000310         AKM(J) = GM(J)/(1.0-2.0*ANUM(J))                                 00020600
000314         IF(ALPM(1))14,14,15
000315      14 WRITE(6,205)(J,EM(J),ANUM(J),GM(J),AKM(J),RHOM(J))               00020800
000335         GO TO 10
000336      15 WRITE(6,206)(J,EM(J),ANUM(J),GM(J),AKM(J),RHOM(J),ALPM(J))       00021000
000360      10 CONTINUE                                                         00021100
         C
000363         IF(ROMS(1).EQ.0)GO TO 777
000364         WRITE (6,217)
000367         DO 779 J=1,NM
         C     CALCULATE THE EFFECTIVE THERMO-ELASTIC PARAMETERS                00021300
         C                                                                      00021400
000371     779 WRITE (6,216) J,ROMS(J)
000403     777 KLINE = 49
000404         L= 0                                                             00021600
000405         DO 100 I=1,NF                                                    00021700
         C     FORM FIBER COMPLIANCE MATRIX
000407         IF(ALPM(1))910,910,911
000410    911  IF(EFT(1).EQ.0)CALL MSUB01(I,EF,EF,ANUF,ANUF,SF)
000415         IF(EFT(1).NE.0)CALL MSUB01(I,EFA,EFT,ANUFA,ANUFT,SF)
000422     910 CONTINUE
000422         DO 100 J=1,NM                                                    00021800
         C     FORM MATRIX COMPLIANCE MATRIX
000424         IF(ALPM(1))912,912,913
000425     913 CALL MSUB01(J,EM,EM,ANUM,ANUM,SM)
000431     912 CONTINUE
000431         DO 100 K=1,NVM                                                   00021900
000433         L= L+1                                                           00022000
000435         MM(L) = J
000437         MF(L) = I
000440         MV(L) = K
000441         VF = 1.0-VM(K)                                                   00022100
```

```
000443          ROCOMP(L)=ROMS(J)*SQRT(3.*(1.+VF)**3/(3.+13.*VF+VF**2+VF**3))
000463          P1 = VM(K)*AKM(J)*(AKF(I)+GM(J))+VF*AKF(I)*(AKM(J)+GM(J))          00022200
000474          Q1 = VM(K)*(AKF(I)+GM(J))+VF*(AKM(J)+GM(J))                        00022300
000505          AKTS(L) = P1/Q1                                                    00022400
000510          IF(EFT(1))1022,1022,1026
000511    1022  CONTINUE
000511          P2 = 4.0*(ANUF(I)-ANUM(J))*(ANUF(I)-ANUM(J))*VF*VM(K)              00022500
000520          Q2 = VM(K)/AKF(I)+VF/AKM(J)+1.0/GM(J)                              00022600
000527          EAS(L) = VF*EF(I)+VM(K)*EM(J)+P2/Q2                                00022700
000537          P3 = VF*VM(K)*(ANUF(I)-ANUM(J))*(1.0/AKM(J)-1.0/AKF(I))            00022800
000551          Q3 = Q2                                                            00022900
000553          ANUAS(L) = VF*ANUF(I)+VM(K)*ANUM(J)+P3/Q3                          00023000
000562          P4 = VM(K)*GM(J)+(1.0+VF)*GF(I)                                    00023100
000571          Q4 = (1.0+VF)*GM(J)+VM(K)*GF(I)                                    00023200
000577          GAS(L) = GM(J)*P4/Q4                                               00023300
000603          GAMMA = GF(I)/GM(J)                                                00023400
000605          BETAM = 1.0/(3.0-4.0*ANUM(J))                                      00023500
000611          BETAF = 1.0/(3.0-4.0*ANUF(I))                                      00023600
000616          GO TO 1028
000616    1026  CONTINUE
000616          P2 = 4.0*(ANUFA(I)-ANUM(J))*(ANUFA(I)-ANUM(J))*VF*VM(K)
000625          Q2 = VM(K)/AKF(I)+VF/AKM(J)+1.0/GM(J)
000634          EAS(L) = VF*EFA(I)+VM(K)*EM(J)+P2/Q2
000644          P3 = VF*VM(K)*(ANUFA(I)-ANUM(J))*(1.0/AKM(J)-1.0/AKF(I))
000656          Q3 = Q2
000660          ANUAS(L) = VF*ANUFA(I)+VM(K)*ANUM(J)+P3/Q3
000667          P4 = VM(K)*GM(J)+(1.0+VF)*GFA(I)
000676          Q4 = (1.0+VF)*GM(J)+VM(K)*GFA(I)
000704          GAS(L) = GM(J)*P4/Q4
000710          GAMMA = GFT(I)/GM(J)
000712          BETAM = 1.0/(3.0-4.0*ANUM(J))
000716          BETAF =  1.0/(1.0 + (2.0*GFT(I))/AKF(I))
000723    1028  CONTINUE
000723          ALEF = 1.E50
000725          IF(GAMMA.NE.1.0) ALEF = (GAMMA+BETAM)/(GAMMA-1.0)
000733          RP = GAMMA*BETAF                                                   00023800
000735          R = (BETAM-RP)/(1.0+RP)                                            00023900
000741          VF3 = VF*VF*VF                                                     00024000
000742          VM2 = VM(K)*VM(K)                                                  00024100
000744          BETAM2 = BETAM*BETAM                                               00024200
000746          X1 = 1.0+R*VF3                                                     00024300
000751          X2 = 3.0*VF*VM2*BETAM2                                             00024400
000754          P5 = (ALEF+BETAM*VF)*X1-X2                                         00024500
000761          Q5 = (ALEF-VF)*X1-X2                                               00024600
000763          IF(GAMMA.EQ.1.0) GTS(L)=GM(J)
000770          IF(GAMMA.NE.1.0) GTS(L)=GM(J)*P5/Q5
000776          P6 = 4.0*AKTS(L)*GTS(L)                                            00024800
001001          Q6 = AKTS(L)+(1.0+4.0*AKTS(L)*ANUAS(L)*ANUAS(L)/EAS(L))*GTS(L)     00024900
001012          ETS(L) = P6/Q6                                                     00025000
001014          ANUTS(L)= 0.5*(ETS(L)/GTS(L))-1.0                                  00025100
001020          IF(ALPM(1))20,20,920
001021     920  IF(EFT(1).NE.0)GO TO 921
001022          BIGKM(J) = EM(J)/(3.0*(1.0-2.0*ANUM(J)))
001030          BIGKF(I) = EF(I)/(3.0*(1.0-2.0*ANUF(I)))                           00025400
001035          BARRK = VM(K)/BIGKM(J)+VF/BIGKF(I)                                 00025500
```

```
001042          ALBAR = VF*ALPF(I)+VM(K)*ALPM(J)
001046          F1 = 0.0
001047          IF(BIGKM(J).NE.BIGKF(I)) F1 = (ALPM(J)-ALPF(I))/
              1 (1.0/BIGKM(J)-1.0/BIGKF(I))
001061          F2 = (3.0*(1.0 - 2.0* ANUAS(L)))/EAS(L)
001067          ALPHA(1,L)= ALBAR + F1*(F2 - BARRK)
001074          ALPHA(2,L)= ALBAR + F1*(3./(2.*AKTS(L)) - ANUAS(L)*F2 - BARRK)
001106          GO TO 20
001106      921 CONTINUE
          C
          C     FORM COMPOSITE COMPLIANCE MATRIX AND CALCULATE MATERIAL THERMAL
          C     EXPANSION COEFFICIENTS
001106          CALL MSUB02(I,J,K,VF,SF,SM)
001112       20 RHOS(L) = VM(K)*RHOM(J)+VF*RHOF(I)
001120          KLINE = KLINE+3
001122          IF (KLINE.LE.48) GO TO 22
001124          KLINE = 0
001125          WRITE(6,215)
001130          IF(ALPM(1))23,23,24
001132       23 WRITE(6,209)
001136          GO TO 22
001137       24 WRITE(6,210)
001143     22   IF(ALPM(1))25,25,26
001145       25 WRITE(6,211)I,J,L,VM(K),EAS(L),ANUAS(L),GAS(L),AKTS(L)
001171          WRITE(6,212)ETS(L),ANUTS(L),GTS(L),RHOS(L)
001205          GO TO 778
001206       26 WRITE(6,213)I,J,L,VM(K),EAS(L),ANUAS(L),GAS(L),AKTS(L),ALPHA(1,L)
001235          WRITE(6,214)ETS(L),ANUTS(L),GTS(L),RHOS(L),ALPHA(2,L)
001254     778  IF(ROMS.NE.0)WRITE(6,218) ROCOMP(L)
001263      100 CONTINUE
          C
001273          GO TO 171
001273     172  WRITE(6,1999)
001277          STOP
001301      200 FORMAT(18I4)
001301      201 FORMAT(1P6E12.5)
001301      202 FORMAT(1H1///42X37HCONSTITUENT THERMO-ELASTIC PARAMETERS)
001301      203 FORMAT(///11H  FIBER NO.,11X4HE(F),12X5HNU(F),11X4HG(F),12X4HK(F),
              111X6HRHO(F),//)
001301      204 FORMAT(///11H  FIBER NO.,11X4HE(F),12X5HNU(F),11X4HG(F),12X4HK(F),
              111X6HRHO(F),9X8HALPHA(F)//)
001301      205 FORMAT(I7,7X,5G16.5)
001301      206 FORMAT(I7,7X,6G16.5)
001301      207 FORMAT(///12H  MATRIX NO.,10X4HE(M),12X5HNU(M),11X4HG(M),
              112X4HK(M),11X6HRHO(M),//)
001301      208 FORMAT(///12H  MATRIX NO.,10X4HE(M),12X5HNU(M),11X4HG(M),
              112X4HK(M),11X6HRHO(M),9X8HALPHA(M)//)
001301      209 FORMAT(///3X1HF,3X1HM,3X8HMATERIAL,6X4HV(M),13X5HE(A)*,10X6HNU(A)*
              1,11X5HG(A)*,11X5HK(T)*,/42X5HE(T)*,10X6HNU(T)*,11X5HG(T)*,
              211X4HRHO*)
001301      210 FORMAT(///3X1HF,3X1HM,3X8HMATERIAL,6X4HV(M),13X5HE(A)*,
              110X6HNU(A)*,11X5HG(A)*,          11X5HK(T)*,9X9HALPHA(A)*,/
              242X5HE(T)*,10X6HNU(T)*,11X5HG(T)*,11X4HRHO*,10X9HALPHA(T)*)
001301      211 FORMAT(/2I4,I7,G15.5,4X,4G16.5)
001301      212 FORMAT(34X,4G16.5)
```

```
001301      213 FORMAT(/2I4,I7,G15.5,4X,5G16.5)                                  00017200
001301      214 FORMAT(34X,5G16.5)                                               00017300
001301      215 FORMAT(1H1///42X35HEFFECTIVE THERMO-ELASTIC PARAMETERS)          00017400
001301      216 FORMAT(I7,7X,G16.5)
001301      217 FORMAT(///12H  MATRIX NO.,8X,8HR-O-S(M)//)
001301      218 FORMAT(/,38X,16H R-O PARAMETER =,1E16.5)
001301     1999 FORMAT(///*+++++++++++++++++++++++++   PROGRAM TERMINATED   ++++
               1++++++++++++++++++++*)
          C
001301          END                                                              00028100
```

```
                 SUBROUTINE MSUB01(I,A,B,C,D,VAR)
             C
             C   ROUTINE MSUB01 FORMS COMPLIANCE MATRIX FOR CONSTITUENTS
             C
             C                                   * VARIABLE DICTIONARY *
             C
             C   A(I)          : LONGITUDINAL MODULUS FOR ITH FIBER/MATRIX
             C   B(I)          : TRANSVERSE MODULUS FOR ITH FIBER/MATRIX
             C   C(I)          : LONGITUDINAL POISSON RATIO FOR ITH FIBER/MATRIX
             C   D(I)          : TRANSVERSE POISSON RATION FOR ITH FIBER/MATRIX
             C   VAR(J,K)      : COMPLIANCE MATRIX
             C
             C                                   * PROGRAMMING INFORMATION *
             C
             C   MSUB01 IS CALLED FROM  : UNI
             C
             C   ************************************************************************
             C
000011           DIMENSION        A(20),B(20),C(20),D(20),VAR(3,3)
000011           VAR(1,1) = 1./A(I)
000013           VAR(2,2) = 1./B(I)
000015           VAR(3,3) = VAR(2,2)
000017           VAR(1,2) = -C(I)/A(I)
000022           VAR(1,3) = VAR(1,2)
000023           VAR(2,3) = -D(I)/B(I)
             C
000026           RETURN
000027           END
```

```
                SUBROUTINE MSUB02(I,J,K,VF,SF,SM)
          C
          C     ROUTINE MSUB02 FORMS COMPOSITE COMPLIANCE MATRIX FOR FIBER BUNDLES
          C     DEFINED IN ROUTINE UNI    AND CALCULATES MATERIAL THERMAL
          C     EXPANSION COEFFICIENTS FOR TRANSVERSELY ISOTROPIC FIBER
          C
          C                                   * VARIABLE DICTIONARY *
          C
          C
          C                                   * PROGRAMMING INFORMATION *
          C
          C     MSUB02 IS CALLED FROM  : UNI
          C
000011          COMMON   /AREA01/EAS(200),ETS(200),ANUAS(200),GAS(200),GTS(200),
               1                  AKTS(200),ANUTS(200)
000011          COMMON   /AREA02/L
000011          COMMON   /AREA03/RHO(200),ALPHA(3,200)
000011          COMMON   /AREA06/EM(20),GM(20),VM(20),ANUM(20),ALPM(20),AKM(20)
000011          COMMON   /AREA07/AKF(20),ALPF(20),ALPFT(20)
000011          COMMON   /AREA10/BIGKM(20),BIGKF(20)
000011          COMMON    /AREA11/MF(200),MV(200),MM(200)
000011          DIMENSION         SF(3,3),SM(3,3),SC(3,3,200)
000011          DIMENSION         SV(3,3),S(3,3),SD(3,3),ABAR(2)
          C
          C     **************************************************************************
          C
          C     CALCULATE ALPHA FOR ANISOTROPIC FIBER CASE
000011          SC(1,1,L) = 1./EAS(L)
000014          SC(2,2,L) = 1./ETS(L)
000017          SC(3,3,L) = SC(2,2,L)
000023          SC(1,2,L) = -ANUAS(L)/EAS(L)
000026          SC(1,3,L) = SC(1,2,L)
000031          SC(2,3,L) = -ANUTS(L)/ETS(L)
          C
000034          EXP1 = SC(1,1,L) - SM(1,1)
000036          EXP2 = SF(2,2)   + SF(2,3)    - SM(2,2)    - SM(2,3)
000043          EXP3 = SM(1,2)   - SC(1,2,L)
000046          EXP4 = SF(1,2)   - SM(1,2)
000051          FXP5 = SF(1,1)   - SM(1,1)
000053          EXP6 = SC(2,2,L) + SC(2,3,L) - SM(2,2)    - SM(2,3)
000061          DAA = ALPF(I) - ALPM(J)
000064          DAT = ALPFT(I) - ALPM(J)
          C
000067          ALPHA(1,L) = (DAA * (EXP1*EXP2 + 2.*EXP3*EXP4)
               *              +2.*DAT * (-EXP1*EXP4 - EXP3*EXP5) ) /
               *              (EXP5*EXP2 - 2.*EXP4*EXP4)
000107          ALPHA(1,L) = ALPHA(1,L) + ALPM(J)
000112          ALPHA(2,L) = (DAA * (-EXP3*EXP2 - EXP4*EXP6) +
               *               DAT * (EXP5*EXP6 + 2.*EXP4*EXP3) ) /
               *              (EXP5*EXP2 - 2.*EXP4*EXP4)
000132          ALPHA(2,L) = ALPHA(2,L) + ALPM(J)
          C
000135          RETURN
000135          END
```

```
              PROGRAM MAIN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
000003        REAL IANG
000003        DIMENSION      E11(20),E22(20),V12(20),V21(20),G12(20),
             1               SCY(20),STY(20),TY(20),
             2               T(20),IANG(20),
             3               S11(20),S12(20),S21(20),S22(20),S44(20),
             4               SINS(20),COSS(20),SIN2(20),COS2(20)
000003        DIMENSION      P11(20),P22(20),P12(20),
             1               EP11(20),EP22(20),EP12(20),
             2               EPS11(20),EPS22(20),EPS12(20)
000003        DIMENSION      BT(60),DC(60),
             1               F(3,20),G(3,20),H(3,20)
000003        DIMENSION      A(60,60),DR(60,60),SG(60,1),SGO(60,1),SF(60),
             1               ULT(6,2,20),
             1               SGS(60),SG1(60,1)
000003        DIMENSION      STRS( 50),STRN( 50),A0(100),POINTS(50,6,20),
             1               STXX(20,50),STYY(20,50),STXY(20,50)
000003        DIMENSION      EXX(50),EYY(50),VXY(50),VYX(50),GXY(50)
000003        DIMENSION      A11(20),A22(20),A44(20),A12(20),B1(20),B2(20)
000003        DIMENSION      EPN(60,1),PS(60)
000003        DIMENSION MATYPE(20),S11T(20),S22T(20),
             1   S11C(20),S22C(20),EP11C(20),EP22C(20),GAMA(20),EP11T(20),
             2   EP22T(20),SIG11(50,20),SIG22(50,20),SIG12(50,20)
          C
000003        COMMON   /SET01/ E11,E22,V12,V21,G12
000003        COMMON   /SET02/ TT,T,IANG
000003        COMMON   /SET03/EP11,EP22,EP12
000003        COMMON   /SET04/ S011,S022,S012,SM11,SM22,SM12
000003        COMMON   /SET05/ ULT,STIFF
000003        COMMON   /SET06/ SIN2,COS2,SINS,COSS
000003        COMMON   /SET07/EPS11,EPS22,EPS12
000003        COMMON   /SET08/ S11,S22,S12,S21
000003        COMMON   /SET09/LAY,EOPT,COPT,IFCN,KSGM,INMT,RATIO,SENS
000003        COMMON   /SET10/ STY,SCY,TY,XM,XN
000003        COMMON   /SET11/ EPS,UPRD,NIT,IT,SMLT
000003        COMMON   /SET12/ F,G,H
000003        COMMON   /SET13/ EXX,EYY,VXY,STXX,STYY,STXY
000003        COMMON   /SET14/ A11,A22,A44,A12,B1,B2
000003        COMMON   /SET15/ POINTS,IPRINT,IOPT,IPTS,LUP
000003        COMMON   /SET16/ MATYPE,S11T,S11C,S22T,S22C,EP11T,EP11C,
             1                 EP22T,EP22C,GAMA,SIG11,SIG22,SIG12
          C
000003        LOGICAL          MSING,MSINGD
000003        INTEGER          EOPT,COPT,SWITCH,AGAIN
000003        INTEGER          UFAIL(3)
          C
000003        EQUIVALENCE      (STXX(1),A0(1))
          C
          C
          C   ARITHMETIC STATEMENT FUNCTIONS
000003        PBO(X,Y,W) = 1.0/(X*Y)**(1./(W-1.))
          C
          C                            * PROGRAM REQUIREMENTS *
          C
```

```
C      SUBROUTINES              : LAMTST,OUTPT1,PROP,REGA1,QUADCF,INVRTD,CMATX,
C                                 CONVR,HEADER ANGLE,MXMULD,TRANS,RESET,NRIRM,
C                                 MATCRL,LAYSUB
C      FUNCTION SUBPROGRAM      : SINT
C      BLOCK DATA SUBPROGRAM    : INITIALIZE VARIABLES IN COMMON SETS
C                                 01,05,10,11, AND 14
C
C                                        * LIMITATIONS *
C
C      LAYERS                   : 20
C      CURVE-FIT DATA POINTS    : 50
C      INCREMENTS               : 50
C
C                                        * DEFAULT VALUES *
C
C      EPS      = 1.00E-03
C      UPBD     = 2.00E+04
C      IT       = 100
C      A12(I)   = 0.00          (FOR ALL LAYERS)
C      XM       = 3.00E 00
C      XN       = 3.00E 00
C      INMT     = 2
C
C
C                                        * INPUT PARAMETERS *
C
C      NST            :NO. OF SEPERATE LAMINATES
C      IOPT           : INPUT OPTION
C                       =1: INPUT RAMBERG-OZGOOD PARAMETERS TY,STY,SCY,XM,XN
C                       =2: DETERMINE RAMBERG-OZGOOD PARAMETERS FROM CURVE-FIT
C      EOPT           : EXPONENT OPTION FOR CURVE-FIT ROUTINE
C                       =1: CURVE-FIT FOR ALL RAMBERG-OZGOOD PARAMETERS
C                       =3: INPUT EXPONENTS, XM + XN, CURVE-FIT TY,STY,SLY
C      COPT           : LAYER OPTION FOR CURVE-FIT ROUTINE
C                       =1: USE SAME STRESS-STRAIN DATA FOR ALL LAYERS
C                       =2: STRESS-STRAIN DATA INPUT FOR EACH LAYER
C      LAMINATE PROPERTIES
C      LAY            : NO. OF LAYERS IN LAMINATE
C      E11(I)         : AXIAL YOUNG#S MODULUS (LAYER I)
C      E22(I)         : TRANSVERSE YOUNG#S MODULUS (LAYER I)
C      V12(I)         : AXIAL-TRANSVERSE POISSON RATIO (LAYER I)
C      G12(I)         : IN-PLANE SHEAR MODULUS (LAYER I)
C      IANG(I)        : ANGULAR ORIENTATION (DEGREES) OF ITH LAYER
C      T(I)           : THICKNESS OF ITH LAYER
C      CURVE-FIT PARAMETERS
C      IPTS           : NO. OF DATA POINTS
C      STRS(I)        : ITH STRESS DATA POINT
C      STRN(I)        : ITH STRAIN DATA POINT
C      POINTS --- COMPONENT
C                     1 - SIGMA 11
C                     2 - EPSILON 11
C                     3 - SIGMA 22
C                     4 - EPSILON 22
C                     5 - SIGMA 12
C                     6 - EPSILON 12
```

```
C       LOADING PARAMETERS
C       S011          : AXIAL APPLIED LOAD
C       S022          : TRANSVERSE APPLIED LOAD
C       S012          : APPLIED SHEAR
C       IFCN          : FAILURE OPTION
C                       =1: ULTIMATE STRESS
C                       =2: QUADRATIC INTERACTION
C                       =3: ULTIMATE STRAIN
C                       =4: ALL FAILURE OPTIONS
C       ULT(I,J,K)    : LIMIT VALUE FOR LAYER K IN DIRECTION I (AXIAL,TRANSVERSE,
C                       OR SHEAR) UNDER J (TENSION OR COMPRESSION)
C       STIFF         : RATIO (TANGENT MODULUS TO INITIAL MODULUS) AT
C                       WHICH COMPUTATIONS TERMINATE
C       A12(I)        : QUADRATIC INTERACTION TERM (LAYER I)
C       CONTROL SENTINELS
C       KSGM          : INCREMENTATION LIMIT
C       SMLT          : MULTIPLICATIVE FACTOR FOR LOAD INCREMENTS
C       IT            : ITERATION LIMIT PER NEWTON-RAPHSON ANALYSIS
C       EPS           : CONVERGENCE CRITERIA
C       UPBD          : DIVERGENCE CRITERIA
C       INMT          : INCREMENTATION ESTIMATE METHOD
C
C
C                                  * VARIABLE DICTIONARY *
C
C
C       A0(I)                     : CURVE-FIT PARAMETER
C       RBO(X,Y,W)                : CONVERSION FUNCTION FROM CURVE-FIT PARAMETERS
C                                   TO RAMBERG-OSGOOD PARAMETERS
C       TT                        : TOTAL THICKNESS
C       SG(I,1)                   : RESULTANT AXIAL STRESS FOR ITH LAYER
C       SG(I+LAY,1)               : RESULTANT TRANSVERSE STRESS FOR ITH LAYER
C       SG(I+2*LAY,1              : RESULTANT SHEAR ITH LAYER
C       SGS(I)                    : AXIAL STRESS (ITH LAYER) FROM PREVIOUS INCREMENT
C       SGS(I+LAY)                : TRANSVERSE STRESS (ITH LAYER) FROM PREVIOUS LOAD
C       SGS(I+2*LAY)              : SHEAR (ITH LAYER) FROM PREVIOUS LOAD
C       SG0(I,1)                  : INITIAL LOAD
C       S11(I)                    : COMPLIANCE TERM
C       S12(I)                    : COMPLIANCE TERM
C       S21(I)                    : COMPLIANCE TERM
C       S22(I)                    : COMPLIANCE TERM
C       S44(I)                    : COMPLIANCE TERM
C       SINS(I)                   : SQUARE OF SIN OF ITH LAYER
C       COSS(I)                   : SQUARE OF COS OF ITH LAYER
C       SIN2(I)                   : TWICE SIN OF ITH LAYER
C       COS2(I)                   : TWICE COS OF ITH LAYER
C       A (I,J)                   : INITIAL TRANSFORMATION MATRIX
C       DB(I,J)                   : MATRIX OF DERIVATIVE TERMS FOR N.-R. APPROX.
C       DC(I)                     : MULTIPLIER OF DERIVATIVE MATRIX  (DB)
C       RT(I)                     : INCREMENTAL CHANGE IN STRESS SOLUTION BETWEEN
C                                   N.-R. ITERATES
C       SG1(I,1)                  : STRESS SOLUTION FROM PREVIOUS NEWTON-RAPHSON
C                                   ITERATION FOR A GIVEN LOAD
C       P11(I)                    : AXIAL STRAIN LAYER AXES (ITH LAYER)
C       P22(I)                    : TRANSVERSE STRAIN LAYER AXES (ITH LAYER)
```

```
         C      P12(I)                   : SHEAR STRAIN LAYER AXES (ITH LAYER)
         C      EP11(I)                  : AXIAL STRAIN LAMINATE AXES (ITH LAYER)
         C      EP22(I)                  : TRANSVERSE STRAIN LAMINATE AXES (ITH LAYER)
         C      EP12(I)                  : SHEAR STRAIN LAMINATE AXES (ITH LAYER)
         C      SF(I)                    : MULTIPLICATION FACTOR USED TO ESTIMATE INITIAL
         C                                 STRESS IN SUCCESSIVE LOADS
         C      PS(I       )             : AXIAL STRAIN LAYER AXES (ITH LAYER) PREVIOUS LOAD
         C      PS(I+  LAY)              : TRAN. STRAIN LAYER AXES (ITH LAYER) PREVIOUS LOAD
         C      PS(I+2*LAY)              : SHEAR STRAIN LAYER AXES (ITH LAYER) PREVIOUS LOAD
         C      EPS11(I)                 : AXIAL STRAIN LAMIN.AXES (ITH LAYER) PREVIOUS LOAD
         C      EPS22(I)                 : TRAN. STRAIN LAMIN.AXES (ITH LAYER) PREVIOUS LOAD
         C      EPS12(I)                 : SHEAR STRAIN LAMIN.AXES (ITH LAYER) PREVIOUS LOAD
         C
         C      SWITCH                   : FAILURE SENTINEL
         C                                 =0: NO FAILURE
         C                                 =1: FAILURE. INTERPOLATE LOAD TO FAILURE POINT
         C      NIT                      : NEWTON-RAPHSON ITERATION COUNTER
         C                 NSR     NON-STANDARD RETURN.  NSR=1 IF AN END-OF-FILE IS TRUE
         C
         C      ************************************************************************
         C      INITIALIZE VARIABLES
         C
000020          EX=3.
000022          INMT=2
000023          RATIO=1.E-06
000024          SENS=1.E-15
         C
         C
         C      PRINT PROGRAM HEADER
000026          CALL HEADER
000027          NST=1
000030      423 CONTINUE
000030          NSR=0
000031          IPRINT=0
000032          CALL MATCRL (NSR)
000033          IF(NSR.EQ.1) GO TO 424
         C      CHANGE ENGINEERING STRAIN TO TENSORIAL STRAIN.
000035      425 DO 63 ILKA=1,50
000037          DO 63 ILKB=1,20
000040       63 POINTS(ILKA,6,ILKB) = POINTS(ILKA,6,ILKB)/2.
         C
         C
         C                               ******************************
         C                               *         INPUT PER SET      *
         C                               ******************************
000050          GO TO (20,30),IOPT
         C      INPUT  1:
000056       20 CONTINUE
000056          GO TO 50
         C      INPUT 2:
000057       30 CONTINUE
000057          IF(COPT.EQ.1)  LUP=1
000062          IF(COPT.EQ.2)  LUP=LAY
000065          DO 40 IL=1,LUP
         C      INPUT STRESS-STRAIN DATA AND FIT CURVE OF FORM:
```

```
       C             STRN = A0 + A1*STRS**EX
       C             WHERE  A0 = A*STRS
000067       DO 40 IC=1,3
000070       DO 35 IDT=1,IPTS
000071       STRN(IDT)=POINTS(IDT,2*IC,LUP)
000077       STRS(IDT)=POINTS(IDT,2*IC-1,LUP)
000104       IF(IC.EQ.1)  A0(IDT) = STRS(IDT)/G12(IL)
000111       IF(IC.EQ.2)  A0(IDT) = STRS(IDT)/E22(IL)
000116       IF(IC.EQ.3)  A0(IDT) = STRS(IDT)/E22(IL)
000123       STRN(IDT) = STRN(IDT) -A0(IDT)
000126       IF(STRN(IDT).LE.1.0E-20)  STRN(IDT) = 0.00
000132    35 CONTINUE
       C     LEAST-SQUARES CURVE-FIT FOR STRESS-STRAIN DATA
000135       CALL REGA1(STRS,STRN,IPTS,EOPT,0.,A1,EX,0)
000144       IF(IC.EQ.1)    TY(IL) = RBO(G12(IL),A1,EX)
000153       IF(IC.EQ.1)        XM = EX
000157       IF(IC.EQ.2)  STY(IL) = RBO(F22(IL),A1,EX)
000165       IF(IC.EQ.2)        XN = EX
000171       IF(IC.EQ.3)  SCY(IL) = RBO(E22(IL),A1,EX)
000177       IF(IC.EQ.3)        XN = EX
000203    40 CONTINUE
000210       IF(COPT.NE.1)  GO TO 50
000212       DO 43 IL=2,LAY
000213       TY(IL)  = TY(1)
000215       STY(IL) = STY(1)
000216       SCY(IL) = SCY(1)
000220    43 CONTINUE
000222    50 CONTINUE
       C     INPUT LOADINGS AND FAILURE CRITERIA
       C     INPUT ANALYSIS CONTROL PARAMETERS
       C     PRINT INPUT
000222       CALL OUTPT1(NST,LAY,IFCN,KSGM)
       C
       C                           ******************************
       C                           *     INITIAL ASSIGNMENTS    *
       C                           *            AND             *
       C                           *       COMPUTATIONS         *
       C                           ******************************
       C     ANGLE  REDUCTION  ROUTINE
000225       CALL ANGLE(LAY,IANG)
       C
000227       SWITCH = 0
000230       TT     =  0.0E0
000231       DO 100 I = 1,LAY
000232       TT      =  TT + T(I)
000234   100 CONTINUE
       C
000236       UFAIL(1) = 0
000237       UFAIL(2) = 0
000240       UFAIL(3) = 0
000241       AGAIN = 0
000242       KSG  =  1
000243       LT1  =  LAY
000244       N    =  LT1
000245       LP1  =  LAY  +1
```

```
000247           LT2  =  LAY*2
000250           LT21 =  LAY*2+1
000251           LT3  =  LAY*3
000252           LM1  =  LAY  -1
           C
000253           DO 105 I=1,LT3
000255           SG(I,1) =  0.000E 00
000256           SGS(I) =  0.0E00
000257           SF(I) =  1.0E0
000261           SG0(I,1) =  0.0E0
000262       105 CONTINUE
           C
000264           DO 107 I = 1,LAY
000265           S11(I) =  1.0E0/E11(I)
000267           S12(I) = -V12(I)/E11(I)
000272           S21(I) = -V21(I)/E22(I)
000274       107 CONTINUE
           C
           C     PRINT INITIAL ELASTIC LAMINATE CONSTANTS
000276           CALL PROP(SG,EXX,EYY,VXY,VYX,GXY,KSGM,KSG,LAY,1)
           C
           C                          ******************************
           C                          *         INCREMENTAL        *
           C                          *           ANALYSIS         *
           C                          ******************************
           C
           C     RETURN TO 110 FOR NEXT INCREMENTAL STEP
000310       110 CONTINUE
000310           MSING = .FALSE.
000311           MSINGD = .FALSE.
000312           NIT  =  0
000313           IPT = 0
000314           SG01 = S011
000315           SG02 = S022
000317           SG03 = S012
           C
000320           DO 111 K = 1,LT3
000322           DC(K)  =  0.0E0
000323           SG1(K,1) =  0.0E0
000324           DO 111 L=1,LT3
000326           DB(K,L)=  0.0E0
000331           A(K,L)   =  0.0E0
000334       111 CONTINUE
000341           DO 1115 I=1,N
000342           S22(I) =  1.0E0/E22(I)
000344           S44(I) =  1.0E0/(4.0E0*G12(I))
000347      1115 CONTINUE
           C
           C     AFTER  2ND  INCREMENTATION  USE  MULTIPLICATIVE  FACTOR
           C     AS  INITIAL  STRESS  SOLUTION  ESTIMATE
000351           IF (KSG.GE.3 .AND. IPT.EQ.0)  GO TO 120
000360       112 CONTINUE
000360           IF(N.EQ.1)  LM1=1
000363           DO 115 I=1,LM1
           C
```

```
000365          SNS  =  SINS(I)
000367          CSS  =  COSS(I)
000370          SN2  =  SIN2(I)
000372          CS2  =  COS2(I)
         C
000373          IF(N.EQ.1) GO TO 113
000375          SNSP =  SINS(I+1)
000377          CSSP =  COSS(I+1)
000400          SN2P =  SIN2(I+1)
000402          CS2P =  COS2(I+1)
         C
000403      113 CONTINUE
000403          A(1,I    ) =  CSS*T(I)
000407          A(1,I+N  ) =  SNS*T(I)
000414          A(1,I+2*N) = -SN2*T(I)
         C
000422          A(2,I    ) =  SNS*T(I)
000426          A(2,I+N  ) =  CSS*T(I)
000432          A(2,I+2*N) =  SN2*T(I)
         C
000440          A(3,I    ) =  SN2*T(I)/2.0E0
000444          A(3,I+N  ) = -SN2*T(I)/2.0E0
000451          A(3,I+2*N) =  CS2*T(I)
000455          IF(N.EQ.1)  GO TO 116
         C
000457          A(3*I+1,I       ) = -S11(I  )*CSS -S21(I  )*SNS
000465          A(3*I+1,I+1     ) =  S11(I+1)*CSSP+S21(I+1)*SNSP
000474          A(3*I+1,I+N     ) = -S12(I  )*CSS -S22(I  )*SNS
000503          A(3*I+1,I+N+1   ) =  S12(I+1)*CSSP+S22(I+1)*SNSP
000512          A(3*I+1,I+2*N   ) =  2.0E0*S44(I  )*SN2
000521          A(3*I+1,I+2*N+1) = -2.0E0*S44(I+1)*SN2P
         C
000530          A(3*I+2,I       ) = -S11(I  )*SNS -S21(I  )*CSS
000536          A(3*I+2,I+1     ) =  S11(I+1)*SNSP+S21(I+1)*CSSP
000545          A(3*I+2,I+N     ) = -S12(I  )*SNS -S22(I  )*CSS
000554          A(3*I+2,I+N+1   ) =  S12(I+1)*SNSP+S22(I+1)*CSSP
000563          A(3*I+2,I+2*N   ) = -2.0E0*S44(I  )*SN2
000572          A(3*I+2,I+2*N+1) =  2.0E0*S44(I+1)*SN2P
         C
000601          A(3*I+3,I       ) = -(S11(I  )-S21(I  ))*SN2 /2.0E0
000610          A(3*I+3,I+1     ) =  (S11(I+1)-S21(I+1))*SN2P/2.0E0
000620          A(3*I+3,I+N     ) = -(S12(I  )-S22(I  ))*SN2 /2.0E0
000627          A(3*I+3,I+N+1   ) =  (S12(I+1)-S22(I+1))*SN2P/2.0E0
000636          A(3*I+3,I+2*N   ) = -2.0E0*S44(I  )*CS2
000644          A(3*I+3,I+2*N+1) =  2.0E0*S44(I+1)*CS2P
         C
000653      115 CONTINUE
         C
000655          A(1,N  ) =  CSSP*T(N)
000661          A(1,2*N) =  SNSP*T(N)
000665          A(1,3*N) = -SN2P*T(N)
000672          A(2,N  ) =  SNSP*T(N)
000676          A(2,2*N) =  CSSP*T(N)
000702          A(2,3*N) =  SN2P*T(N)
000706          A(3,N  ) =  SN2P*T(N)/2.0E0
```

```
000712         A(3,2*N) = -SN2P*T(N)/2.0E0
000716         A(3,3*N) =  CS2P*T(N)
000723     116 CONTINUE
000723         IF(MSING)  GO TO 117
         C
         C
         C
         C     INVERT  MATRIX  A
000725         CALL INVRTD(A,60,LT3,DET, SENS  ,IRANK,1.00E-30)
         C
         C     CHECK  FOR  SINGULAR  MATRIX
000733         IF(IRANK.EQ.LT3)  GO TO 118
000735         MSING = .TRUE.
000736         WRITE(6,1420) IRANK,DET
000745         GO TO 112
000746     117 CONTINUE
000746         WRITE(6,1425) ((A(K1,L1),L1=1,LT3),K1=1,LT3)
000766         MSING = .FALSE.
000767         GO TO 999
000770     118 CONTINUE
         C
         C     INITIAL LOAD VECTOR
000770         IF (SWITCH.EQ.1)  GO TO 119
000772         SGO(1,1) =  SO11*TT
000774         SGO(2,1) =  SO22*TT
000776         SGO(3,1) =  SO12*TT
000777     119 CONTINUE
000777         CALL MXMULD(A,SGO,SG,60,60,1,LT3,LT3,1)
         C
         C     RESET  STRESS = 0,  IF  RELATIVE  STRESS ) 1.0D-06
001010         CALL RESET(LT3,SG, RATIO )
001013         GO TO 126
         C
         C     (MULTIPLICATIVE FACTOR) X (SOLUTION FROM PREVIOUS INCREMENT)
001014     120 CONTINUE
001014         DO 122 I=1,LT3
001016         SG(I,1) =  SF(I)*SGS(I)
001021     122 CONTINUE
001023         GO TO 126
         C
         C     RETURN  TO  125  FOR  NEXT  ITERATION  STEP
001023     125 CONTINUE
001023         NIT =  NIT + 1
001025     126 CONTINUE
         C     STORE STRESS SOLUTION FOR THIS ITERATION
001025         DO 127 I=1,LT3
001027         SG1(I,1) =  SG(I,1)
001031         BT(I) =  0.0E 00
001032     127 CONTINUE
         C
         C
001034     130 CONTINUE
001034         DO 1305 K=1,LT3
001036         DO 1305 L=1,LT3
001037         DB(K,L) =  0.0E 00
```

```
001042     1305 CONTINUE
          C                            ********************************
          C                            *    DERIVATIVE MATRIX         *
          C                            *           FOR                *
          C                            *    NEWTON-RAPHSON ANALYSIS   *
          C                            ********************************
001046          IF(N.EQ.1) LM1=1
001051          DO 151 I=1,LM1
          C     CALCULATION OF TERMS USED IN FORMATION OF DERIVATIVE MATRIX
001053          IF(N.EQ.1)  GO TO 1308
001055          CALL  NRTRM (LAY,SG,F,G,H,I)
001061          CALL  NRTRM (LAY,SG,F,G,H,I+1)
001067     1308 CONTINUE
          C
001067          SNS  =  SINS(I)
001071          CSS  =  COSS(I)
001073          SN2  =  SIN2(I)
001074          CS2  =  COS2(I)
          C
001076          IF(N.EQ.1)  GO TO 131
001100          SNSP =  SINS(I+1)
001101          CSSP =  COSS(I+1)
001103          SN2P =  SIN2(I+1)
001104          CS2P =  COS2(I+1)
          C
001106      131 CONTINUE
001106          DB(1,I)        =  CSS*T(I)
001112          DB(1,I+N)      =  SNS*T(I)
001117          DB(1,I+2*N)    = -SN2*T(I)
001125          DB(2,I)        =  SNS*T(I)
001131          DB(2,I+N)      =  CSS*T(I)
001135          DB(2,I+2*N)    =  SN2*T(I)
001143          DB(3,I)        =  SN2*T(I)/2.
001147          DB(3,I+N)      = -SN2*T(I)/2.
001154          DB(3,I+2*N)    =  CS2*T(I)
001160          IF(N.EQ.1)  GO TO 161
          C
001162          DB(3*I+1,I)           = -F(1,I)
001167          DB(3*I+1,I+1)         =  F(1,I+1)
001174          DB(3*I+1,I+N)         = -F(2,I)
001201          DB(3*I+1,I+N+1)       =  F(2,I+1)
001207          DB(3*I+1,I+2*N)       = -F(3,I)
001215          DB(3*I+1,I+2*N+1)     =  F(3,I+1)
          C
001223          DB(3*I+2,I)           = -G(1,I)
001231          DB(3*I+2,I+1)         =  G(1,I+1)
001236          DB(3*I+2,I+N)         = -G(2,I)
001243          DB(3*I+2,I+N+1)       =  G(2,I+1)
001251          DB(3*I+2,I+2*N)       = -G(3,I)
001260          DB(3*I+2,I+2*N+1)     =  G(3,I+1)
          C
001266          DB(3*I+3,I)           = -H(1,I)
001274          DB(3*I+3,I+1)         =  H(1,I+1)
001301          DB(3*I+3,I+N)         = -H(2,I)
001306          DB(3*I+3,I+N+1)       =  H(2,I+1)
```

```
001314          DB(3*I+3,I+2*N)      = -H(3,I)
001323          DB(3*I+3,I+2*N+1)    =  H(3,I+1)
          C
001331      151 CONTINUE
001334          DB(1,N)        =  CSSP*T(N)
001340          DB(1,2*N)      =  SNSP*T(N)
001344          DB(1,3*N)      = -SN2P*T(N)
001351          DB(2,N)        =  SNSP*T(N)
001355          DB(2,2*N)      =  CSSP*T(N)
001361          DB(2,3*N)      =  SN2P*T(N)
001365          DB(3,N)        =  SN2P*T(N)/2.
001371          DB(3,2*N)      = -SN2P*T(N)/2.
001375          DB(3,3*N)      =  CS2P*T(N)
001402      161 CONTINUE
001402          IF(MSINGD)  GO TO 167
          C
          C     INVERT  MATRIX  DB
001404          CALL INVRTD(DB,60,LT3,DET, SENS  ,IRANK,1.00E-30)
          C
          C     CHECK  FOR  SINGULAR  MATRIX
001412          IF(IRANK.EQ.LT3)  GO TO 168
001414          MSINGD = .TRUE.
001415          WRITE(6,1420) IRANK,DET
001424          GO TO 130
001425      167 CONTINUE
001425          WRITE(6,1425) ((DB(K1,L1),L1=1,LT3),K1=1,LT3)
001445          MSINGD = .FALSE.
001446          GO TO 999
001447      168 CONTINUE
          C
          C                             ******************************
          C                             *        INITIAL VECTOR      *
          C                             *             FOR            *
          C                             *   NEWTON-RAPHSON ANALYSIS  *
          C                             ******************************
          C
          C
001447          IF (SWITCH.EQ.1)  GO TO 173
001451          DC(1)         = -S011*TT
001453          DC(2)         = -S022*TT
001455          DC(3)         = -S012*TT
001456          GO TO 174
001457      173 CONTINUE
001457          DC(1)         = -SG0(1,1)
001461          DC(2)         = -SG0(2,1)
001462          DC(3)         = -SG0(3,1)
001464      174 CONTINUE
          C
001464          DO 6 I=1,LAY
001466          SNS  =  SINS(I)
001470          CSS  =  COSS(I)
001471          SN2  =  SIN2(I)
001473          CS2  =  COS2(I)
001474          DC(1)         =  DC(1) + SG(I,1)*CSS*T(I) + SG(LAY+I,1)*SNS*T(I)
               1                 -SG(2*LAY+I,1)*SN2*T(I)
```

```
001510          DC(2)         =  DC(2) + SG(I,1)*SNS*T(I) + SG(LAY+I,1)*CSS*T(I)
               1                 +SG(2*LAY+I,1)*SN2*T(I)
001524          DC(3)         =  DC(3) + SG(I,1)*T(I)*SN2/2. - SG(LAY+I,1)*T(I)*
               1                 SN2/2. + SG(2*LAY+I,1)*CS2*T(I)
001540        6 CONTINUE
        C
        C       FORM NON-LINEAR COMPLIANCE TERMS S22 AND S44
001543          DO 7 K=1,N
001544          T12S =  (SG(K+2*N,1)/TY(K))**2
001550          IF(SG(K+N,1)-0.0) 1,2,2
001553        1 TS22S =(SG(K+N,1)/SCY(K))**2
001557          GO  TO  3
001557        2 TS22S =(SG(K+N,1)/STY(K))**2
001563        3 S12S  = T12S +TS22S
001565          S22(K) =  (1.0E0+S12S**((XN-1.)/2.))/E22(K)
001577          S44(K) =  (1.0E0+S12S**((XM-1.)/2.))/(4.0E0*G12(K))
001612        7 CONTINUE
        C
001614          IF(N.EQ.1)  GO TO 5008
001616          DO 8 K=1,LM1
001617          SNS  =  SINS(K)
001621          CSS  =  COSS(K)
001622          SN2  =  SIN2(K)
001624          CS2  =  COS2(K)
001625          SNSP =  SINS(K+1)
001627          CSSP =  COSS(K+1)
001630          SN2P =  SIN2(K+1)
001632          CS2P =  COS2(K+1)
        C
001633          DC(3+3*K-2)       =  -(S11(K)*CSS+S21(K)*SNS)*SG(K,1) -
               1          SG(LAY+K,1)*(S12(K)*CSS+S22(K)*SNS) + 2.*S44(K)*SN2*
               2                      SG(2*LAY+K,1) + (S11(K+1)*CSSP+S21(K+1)*SNSP)
               3                     *SG(K+1,1) + (S12(K+1)*CSSP+S22(K+1)*SNSP)*
               4                      SG(LAY+K+1,1) - 2.*S44(K+1)*SG(2*LAY+K+1,1)*
               5                      SN2P + SG0(3+3*K-2,1)
001706          DC(3+3*K-1)   = -(S11(K)*SNS+S21(K)*CSS)*SG(K,1) - (S12(K)*SNS+
               1                 S22(K)*CSS)*SG(LAY+K,1) - 2.*S44(K)*SN2*
               2                 SG(2*LAY+K,1) +
               3                (S11(K+1)*SNSP+S21(K+1)*CSSP)*SG(K+1,1) + (S12(K+1)*
               4                 SNSP+S22(K+1)*CSSP)*SG(LAY+K+1,1)+ 2.*S44(K+1)*SN2P
               5                *SG(2*LAY+K+1,1) + SG0(3+3*K-1,1)
001761          DC(3+3*K)     =  -(S11(K)-S21(K))*SN2*SG(K,1)/2. -(S12(K)-S22(K))*
               1                 SN2*SG(LAY+K,1)/2. - 2.*S44(K)*CS2*SG(2*LAY+K,1) +
               2                 (S11(K+1)-S21(K+1))*SN2P*SG(K+1,1)/2. + (S12(K+1)-
               3                 S22(K+1))*SN2P*SG(LAY+K+1,1)/2. + 2.*S44(K+1)*CS2P*
               4                 SG(2*LAY+K+1,1) + SG0(3+3*K,1)
002035        8 CONTINUE
002037     5008 CONTINUE
        C
002037          DO 9 I=1,LT3
002041          DC(I) = -DC(I)
002043        9 CONTINUE
        C
002045          DO 11 I=1,LT3
002046          DO 11 K=1,LT3
```

```
002047       11 BT(I) =  BT(I) + DB(I,K)*DC(K)
          C
          C                              ********************************
          C                              *      FORM SOLUTION VECTOR    *
          C                              *               FOR            *
          C                              *         THIS ITERATION       *
          C                              ********************************
002062          DO 15 I=1,LT3
002063          SG(I,1) =  SG1(I,1) + BT(I)
002066       15 CONTINUE
          C
          C     RESET  STRESS = 0,  IF  RELATIVE  STRESS ) 1.0D-06
002070          CALL RESET(LT3,SG, RATIO )
002073          IF(NIT.EQ.0)  GO TO 125
          C
          C                              ********************************
          C                              *        CONVERGENCE CHECK     *
          C                              ********************************
002074          CALL CONVR(LAY,SG,SG1,KSG,IRTN)
002100          GO TO (495,125,900),IRTN
002107      495 CONTINUE
002107          IF(SWITCH.EQ.0)  GO TO 500
002110          SG01 = SG0(1,1)/TT
002112          SG02 = SG0(2,1)/TT
002114          SG03 = SG0(3,1)/TT
002115      500 CONTINUE
          C
          C                              ********************************
          C                              *      STRAIN COMPUTATIONS     *
          C                              ********************************
002115          DO 540 I = 1,LAY
002117          SNS  =  SINS(I)
002121          CSS  =  COSS(I)
002122          SN2  =  SIN2(I)
002124          CS2  =  COS2(I)
          C
          C     STRAIN COMPUTATIONS IN FIBER AXES DIRECTIONS
002125          P11(I) =  S11(I)*SG(I,1) + S12(I)*SG(I+N,1)
002132          T12S =  (SG(I+2*N,1)/TY(I))**2
002136          IF(SG(I+N,1)-0.0) 4,5,5
002142        4 TS22S =(SG(I+N,1)/SCY(I))**2
002146          GO TO 18
002146        5 TS22S=(SG(I+N,1)/STY(I))**2
002152       18 S12S=T12S+TS22S
002154          P22(I) =  S21(I)*SG(I,1) + SG(I+N,1)/E22(I)*(1.0E0+S12S**((XN
              1           -1.)/2.))
002173          P12(I) =  SG(I+2*N,1)/(2.*G12(I))*(1.0E0+S12S**((XM-1.)/2.))
002211      520 CONTINUE
002211          EPN(I    ,1) = P11(I)
002213          EPN(I+  N,1) = P22(I)
002216          EPN(I+2*N,1) = P12(I)
          C
          C     STRAIN COMPUTATIONS IN LAMINATE AXES DIRECTIONS
002221          EP11(I) =  P11(I)*CSS + P22(I)*SNS - P12(I)*SN2
002227          EP22(I) =  P11(I)*SNS + P22(I)*CSS + P12(I)*SN2
```

```
002235          FP12(I) =  (P11(I)-P22(I))*SN2/2.E0 + P12(I)*CS2
              C
002243        540 CONTINUE
002246          IF (AGAIN.EQ.1)  GO TO 752
002250          IF (SWITCH.EQ.0)  GO TO 610
002251          SWITCH = 0
002251          IPT = 1
002252          GO TO 730
              C
              C                           ******************************
              C                           *    LAMINATE FAILURE TESTS   *
              C                           ******************************
002253        610 CONTINUE
002253          CALL LAMTST(LAY,SG,SGS,EPN,PS,KSG,KSGM,IFCN,UFAIL,FAC,SWITCH)
              C
              C                           ******************************
              C                           *         OUTPUT PER          *
              C                           *       LOAD INCREMENT        *
              C                           ******************************
002266        730 CONTINUE
002266          WRITE(6,1525)
002272          WRITE(6,1527) SG01
002300          WRITE(6,1528) SG02
002306          WRITE(6,1529) SG03
002314          WRITE(6,1735) NIT
002322          WRITE(6,1536)
002326          WRITE(6,1537)
002332          WRITE(6,1538)
002336          DO 750 I = 1,LAY
002340          EP12(I)=EP12(I)*2.
002342          WRITE(6,1550) I,SG(I,1),SG(I+N,1),SG(I+2*N,1),
               1                EP11(I),EP22(I),EP12(I),P11(I),P22(I),P12(I)
002375          EP12(I)=EP12(I)/2.
002400        750 CONTINUE
002402          IF(IPT.EQ.1)  WRITE(6,1990)
002407          IF (IPT.EQ.1)  AGAIN=1
002412          IF(IPT.EQ.1)  GO TO 110
              C
              C COMPUTE INELASTIC MATERIAL PROPERTIES AND RETAIN AS FUNCTION OF INCREMENT
002413        752 CONTINUE
002413          AGAIN = 0
002414          CALL PROP(SG,EXX,EYY,VXY,VYX,GXY,KSGM,KSG,LAY,2)
002426          DO 755 I=1,LAY
002430          STXX(I,KSG) = SG(I     ,1)
002434          STYY(I,KSG) = SG(I+   N,1)
002441          STXY(I,KSG) = SG(I+2*N,1)
002446        755 CONTINUE
              C
              C CHECK FOR LAMINATE FAILURE, IF FAILURE HAS OCCURED INTERPOLATE LOADS TO
              C FAILURE POINT AND REEVALUATE
002450          IF (SWITCH.EQ.0)  GO TO 758
002451          WRITE(6,1555)
002455          WRITE(6,1560)
002461          SG0(1,1) = ((S011-SM11) + FAC*SM11)*TT
002466          SG0(2,1) = ((S022-SM22) + FAC*SM22)*TT
```

```
002472          SGO(3,1) = ((SO12-SM12) + FAC*SM12)*TT
002477          GO TO 110
002477      758 CONTINUE
          C
          C     CHECK FOR INCREMENTATION LIMIT
002477          IF(KSGM-KSG) 790,790,760
          C
          C                        ******************************
          C                        *          INCREMENTATION    *
          C                        *          ESTIMATE          *
          C                        ******************************
          C
002502      760 CONTINUE
002502          GO TO (762,766), INMT
          C
          C     RATIO  OF  PREVIOUS  SOLUTIONS
002510      762 CONTINUE
002510          DO 765 I=1,LT3
002512          IF(KSG.EQ.1)  GO TO 770
002514          IF(SGS(I).EQ.0.0E0)  GO TO 763
002515          SF(I) =  SG(I,1)/SGS(I)
002517          GO TO 765
002520      763 CONTINUE
002520          SF(I) =  1.0E0
002522      765 CONTINUE
002525          GO TO 770
          C
002525      766 CONTINUE
002525          DO 768 I=1,LT3
002527          IF(KSG.LT.2)  GO TO 767
002531          IF(SG(I,1).EQ.0.0E0)  GO TO 767
002533          VKSG =  KSG + 1
002535          CONS  =  VKSG*(VKSG-2)/(VKSG-1)**2
002543          SF(I) =  1.0E0 + CONS*(SG(I,1)-SGS(I))/SG(I,1)
002551          GO TO 768
002551      767 CONTINUE
002551          SF(I) =  1.0E0
002553      768 CONTINUE
          C
          C     STORE  STRESS  AND  STRAIN  VALUES
002556      770 CONTINUE
002556          DO 775 I = 1,LAY
002560          SGS(I    ) =  SG(I    ,1)
002562          SGS(I+  N) =  SG(I+  N,1)
002566          SGS(I+2*N) =  SG(I+2*N,1)
002572          EPS11(I) =  EP11(I)
002573          EPS22(I) =  EP22(I)
002575          EPS12(I) =  EP12(I)
002576          PS(I    ) = P11(I)
002600          PS(I+  N) = P22(I)
002602          PS(I+2*N) = P12(I)
002605      775 CONTINUE
          C
          C     INCREMENT  APPLIED  LOADING
002610          KSG  =  KSG + 1
```

```
RUN VERSION 2.3 --PSR LEVEL 363--                                MAIN                                  07/23/74
002611            SO11 =  SO11 + SM11
002613            SO22 =  SO22 + SM22
002615            SO12 =  SO12 + SM12
002617            GO TO 110
002620        790 CONTINUE
            C
            C     LEAST-SQUARES CURVE-FIT OF FXX, EYY, AND VXY AGAINST SXX, SYY, AND SXY
002620        900 CONTINUE
002620            NST=NST+1
            C
002622        999 CONTINUE
002622            CALL LAYSUB (A12,STY,SCY,TY,EPS11,EPS22,EPS12)
002631            GO TO 423
002632        424 WRITE(6,1999)
002636            STOP
            C
            C
            C
            C
            C
            C
002640       1420 FORMAT (//* MATRIX  IS  SINGULAR  OF  RANK  =  *,I2,
                 *         *+   DETERMINANT  =  *,E15.5/)
002640       1425 FORMAT (6E15.5)
002640       1525 FORMAT (* EXTERNAL APPLIED STRESS*/)
002640       1527 FORMAT (* SG XX  = *,E13.5)
002640       1528 FORMAT (* SG YY  = *,E13.5)
002640       1529 FORMAT (* SG XY  = *,E13.5)
002640       1536 FORMAT (/22X,*STRESS*,38X,*STRAIN*,37X,*STRAIN*)
002640       1537 FORMAT (19X,*(LAYER AXES)*,31X,*(LAMINATE AXES)*,
                 *         29X,*(LAYER AXES)*)
002640       1538 FORMAT ( /* LAYER*,4X,*SG 11*,8X,*SG 22*,8X,*SG 12 *,11X,*EP XX*,
                 1         8X,*EP YY*,8X,*EP XY *,11X,*EP 11*,8X,*EP 22*,8X,
                 2         *EP 12 */)
002640       1550 FORMAT (I4,1X,2(3E13.5,4X),3E13.5)
002640       1555 FORMAT (////40X,*****  LAMINATE ANALYSIS INTERPOLATED TO *,
                 *         *FAILURE POINT  *****//)
002640       1560 FORMAT (* AT  FAILURE*)
002640       1735 FORMAT (/* SOLUTION  FOR  STRESS  CONVERGES  WITHIN*,I4,
                 *         *  ITERATIONS*/)
002640       1990 FORMAT (////)
002640       1999 FORMAT(///*++++++++++++++++++++++++   PROGRAM TERMINATED   ++++++
                 1++++++++++++++++++++*)
002640            END
```

```
                SUBROUTINE HEADER
          C
          C     ROUTINE HEADER PRINTS HEADER INFORMATION FOR NOLIN V2 M2
          C
000002          REAL            MESSAG(100)
          C     RETRIEVE JULIAN DATE FROM THE OPERATING SYSTEM.....
000002          DATE=0
          C     READ IN 5 CARD PROGRAM IDENTIFICATION
000003          READ(5,200) (MESSAG(I),I=1,100)
000015      200 FORMAT(20A4)
          C     WRITE OUT TITLE, DATE, AND PROGRAM IDENTIFICATION
000015          WRITE(6,1000) DATE, (MESSAG(J),J=1,100)
000031     1000 FORMAT (1H1,///,41X,41(*+*),3(/,41X,*+*,39X,*+*),/,
               *         41X,*+*,14X,*NONLINEAR*,16X,*+*,/,41X,*+*,39X,*+*,/,
               *         41X,*+*, 8X,*THERMOELASTIC ANALYSIS*, 9X,*+*,/,
               *         41X,*+*,39X,*+*,/,41X,*+*,18X,*OF*,19X,*+*,/,
               *         41X,*+*,39X,*+*,/,41X,*+*,10X,*FIBROUS COMPOSITES*,
               *         11X,*+*,/,41X,*+*,39X,*+*,/,41X,*+*,17X,*AND*,19X,*+*,/,
               *         41X,*+*,39X,*+*,/,41X,*+*, 6X,
               *         *NON-HOMOGENEOUS LAMINATES*, 8X,*+*,3(/,41X,*+*,39X,*+*),
               *         /,41X,41(*+*),//,
               *         //21X,*.VERSION 2     MOD 3  (MAY 74)*,
               *                         //,21X,*.DATE        *,A10,///,21X,
               *         *.PROGRAM IDENTIFICATION*,//,5(21X,20A4,/),///)
          C
000031          RETURN
000032          END
```

```
                   SUBROUTINE MATCRL (NSR)
000003             REAL THETA
000003             DIMENSION THETA(20),THICK(20),MATYPE(20),S11T(20),S22T(20),
                  1          S12(20),S11C(20),S22C(20),EP11T(20),EP22T(20),EP11C(20)
                  2          ,EP22C(20),GAMA(20),E11(20),E22(20),G12(20),V12(20),
                  3          T(20),A12(20),V21(20),TE11(20),TE22(20),TG12(20)
                  4          ,TV12(20),TA12(20),TSTY(20),TSCY(20),TTY(20),TY(20),
                  5          STY(20),SCY(20),EP11(20),EP22(20),EP12(20),ULT(6,2,20),
                  6          POINTS(50,6,20),EPS11(50,20),EPS22(50,20),EPS12(50,20),
                  7          TEPS11(50,20),TEPS22(50,20),TEPS12(50,20),
                  8         SIG11(50,20),SIG22(50,20),SIG12(50,20),A11(20),A22(20),
                  9          A44(20),B1(20),B2(20),S11(20),S22(20),S21(20)
000003             COMMON   /SET01/ E11,E22,V12,V21,G12
000003             COMMON   /SET02/TT,THICK,THETA
000003             COMMON   /SET03/EP11,EP22,EP12
000003             COMMON   /SET04/ SO11,SO22,SO12,SM11,SM22,SM12
000003             COMMON   /SET05/ ULT,STIFF
000003             COMMON   /SET08/ S11,S22,S12,S21
000003             COMMON   /SET09/ NLAY,EOPT,COPT,IFCN,KSGM,INMT,RATIO,SENS
000003             COMMON   /SET10/ STY,SCY,TY,XM,XN
000003             COMMON   /SET11/ EPS,UPBD,NIT,IT,SMLT
000003             COMMON   /SET14/ A11,A22,A44,A12,B1,B2
000003             COMMON   /SET15/ POINTS,IPRINT,IOPT,IPTS,LUP
000003             COMMON   /SET16/ MATYPE,S11T,S11C,S22T,S22C,EP11T,EP11C,
                  1                 EP22T,EP22C,GAMA,SIG11,SIG22,SIG12
000003             COMMON   /SET17/ TE11,TE22,TG12,TV12,TA12,TSTY,TSCY,TTY
000003             INTEGER  EOPT,COPT
000003             NAMELIST/DATA/NLAY,E11,E22,V12,G12,THICK,THETA,IOPT,
                  1   STY,SCY,TY,XM,XN,EOPT,COPT,IPTS,IPRINT,SO11,SO22,SO12,
                  2   IFCN,STIFF,A12,KSGM,SMLT,IT,EPS,UPBD,INMT,RATIO,SENS,
                  3   S11T,S22T,S12,S11C,S22C,EP11T,EP22T,EP11C,EP22C,
                  4   GAMA,SIG11,SIG22,SIG12,EPS11,EPS22,EPS12,MATYPE
000003             NSR=0
000004             READ(5,DATA)
000006             IF(EOF,5) 424,425
000012         425 DO 10 I=1,20
000014             TE11(I) = E11(I)
000016             TE22(I) = E22(I)
000017             TG12(I) = G12(I)
000021             TV12(I) = V12(I)
000022             TA12(I) = A12(I)
000024             TSTY(I) = STY(I)
000025             TSCY(I)= SCY(I)
000027             TSCY(I) = SCY(I)
000030          10 TTY(I)  = TY(I)
000033             DO 200 I = 1,20
000035             M = MATYPE(I)
000037             E11(I) = TE11(M)
000041             E22(I) = TE22(M)
000042             G12(I) =TG12(M)
000044             V12(I) = TV12(M)
000045             A12(I) = TA12(M)
000047             STY(I) = TSTY(M)
000050             SCY(I) = TSCY(M)
```

```
)00052          TY(I) = TTY(M)
)00053          ULT(1,1,I) = S11T(M)
)00056          ULT(1,2,I) = S11C(M)
)00061          ULT(2,1,I) = S22T(M)
)00064          ULT(2,2,I) = S22C(M)
)00067          ULT(3,1,I) = S12(M)
)00072          ULT(3,2,I) = S12(M)
)00075          ULT(4,1,I) = EP11T(M)
)00100          ULT(4,2,I) = EP11C(M)
)00103          ULT(5,1,I) = EP22T(M)
)00106          ULT(5,2,I) = EP22C(M)
)00111          ULT(6,1,I) = GAMA(M)
)00114          ULT(6,2,I) = GAMA(M)
)00117          DO 200 J = 1,IPTS
)00121          POINTS(J,1,I) = SIG11(J,M)
)00127          POINTS(J,2,I) = EPS11(J,M)
)00135          POINTS(J,3,I) = SIG22(J,M)
)00143          POINTS(J,4,I) = EPS22(J,M)
)00151          POINTS(J,5,I) = SIG12(J,M)
)00157      200 POINTS(J,6,I) = EPS12(J,M)
)00171          RETURN
)00172      424 NSR=1
)00173          RETURN
)00174          END
```

```
                SUBROUTINE OUTPT1(NST,LAY,IFCN,KSGM)
          C
          C     ROUTINE OUTPT1 PRINTS OUT ALL PARAMETERS PERTAINING
          C     TO PROBLEM DEFINITION
          C
00007           REAL IANG
00007           DIMENSION         E11(20),E22(20),V12(20),V21(20),G12(20),
               1                  SCY(20),STY(20),TY(20),
               2                  T(20),IANG(20)
00007           DIMENSION         ULT(6,2,20)
00007           DIMENSION         HDFAIL(8),POINTS (50,6,20)
00007           DIMENSION         A11(20),A22(20),A44(20),A12(20),B1(20),B2(20)
00007           COMMON   /SET01/  E11,E22,V12,V21,G12
00007           COMMON   /SET02/  TT,T,IANG
00007           COMMON   /SET04/  SO11,SO22,SO12,SM11,SM22,SM12
00007           COMMON   /SET05/  ULT,STIFF
00007           COMMON   /SET10/  STY,SCY,TY,XM,XN
00007           COMMON   /SET11/  EPS,UPHD,NIT,IT,SMLT
00007           COMMON   /SET14/  A11,A22,A44,A12,B1,B2
00007           COMMON   /SET15/  POINTS,IPRINT,IOPT,IPTS,LUP
00007           DATA              HDFAIL/10HULTIMATE S,10HTRESS     ,
               *                         10HULTIMATE S,10HTRESS     ,
               *                         10HQUAD. INTE,10HRACTION   ,
               *                         10HALL FAILUR,10HE CRITERIA/
          C
00007           SM11 = SMLT*SO11
00011           SM22 = SMLT*SO22
00013           SM12 = SMLT*SO12
00015           DO 15 I=1,LAY
00016           V21(I) = V12(I)*E22(I)/E11(I)
00022        15 CONTINUE
          C
00024           WRITE(6,1509)  NST
00031           WRITE(6,1510) LAY
00042           IF(IPRINT.NE.1) GO TO 1112
00047           WRITE(6,1507)
00052           DO 47 IL = 1,LUP
00056           DO 45 J = 1,5
00057        45 WRITE(6,1553)  (POINTS(I,J,IL),I=1,IPTS)
00102           DO 46 IAC = 1,IPTS
00103        46 POINTS(IAC,6,IL) = POINTS(IAC,6,IL)*2.
00111           WRITE(6,1553)  (POINTS(I,6,IL),I=1,IPTS)
00126           DO 47 IAD = 1,IPTS
00132        47 POINTS(IAD,6,IL) = POINTS(IAD,6,IL)/2.
00142      1112 WRITE(6,1513)
00146           DO 60 I=1,LAY
00152           WRITE(6,1515)   I,IANG(I),T(I),E11(I),E22(I),V12(I),V21(I),
               1                G12(I),STY(I),SCY(I),TY(I)
00203        60 CONTINUE
00210           WRITE(6,1516)
00213           WRITE(6,1517) XM
00221           WRITE(6,1518) XN
00227           WRITE(6,1520)
00233           WRITE(6,1524) SO11,SM11,KSGM
```

```
00247          WRITE(6,1526) SO22,SM22
00257          WRITE(6,1528) SO12,SM12
00267          IFN = 2*IFCN
00273          IST = IFN-1
00275          WRITE(6,1530) (HDFAIL(IF),IF=IST,IFN)
00307          IF(IFCN.NE.2)  WRITE(6,1532)
00322          IF((IFCN.NE.2).OR.(IFCN.EQ.4))  IFS=1
00332          IF((IFCN.EQ.2).OR.(IFCN.EQ.4)) WRITE(6,1533)
00350          IF(IFCN.FQ.2)  IFS=4
00353          DO 90 IL=1,LAY
00355       85 CONTINUE
00355          IFE = IFS+2
00357          WRITE(6,1534)  IL,((ULT(ID,IM,IL),ID=IFS,IFE),IM=1,2)
00404          IF(IFCN.NE.4)  GO TO 90
00411          IF(IFS.EQ.1)  IFS=4
00414          IF(IFE.FQ.3)  GO TO 85
00416          IFS = 1
00417       90 CONTINUE
00422          IF((IFCN.EQ.1).OR.(IFCN.EQ.2)) GO TO 110
00431          WRITE(6,1535)
00435          DO 95 IL=1,LAY
00441          WRITE(6,1536)  IL,A12(IL)
00450       95 CONTINUE
00455      110 CONTINUE
00455          WRITE(6,1538) STIFF
00463          WRITE(6,1540)
00467          WRITE(6,1544) IT
00475          WRITE(6,1546) EPS
00503          WRITE(6,1548) UPBD
        C
        C
00511     1507 FORMAT(///* DATA INPUT POINTS FOR CURVE FIT-*///)
00511     1509 FORMAT (1H1,50X,*LAMINATE*,I4,/50X,12(*+*))
00511     1510 FORMAT (////* NUMBER  OF  LAYERS = *,I2)
00511     1513 FORMAT (/* LAYER   THETA*,6X,*T*,13X,*E11*, 9X,*E22*, 9X,
              *        *V12*, 9X,*V21*, 9X,*G12*,8X,*SGT Y*,8X,*SGC Y*,8X,*TAUY*/)
00511     1515 FORMAT(I4,3X,F6.2,2X,E11.4,4X,8(1X,E11.4))
00511     1516 FORMAT (//* EQUATION  PARAMETERS*)
00511     1517 FORMAT (/*    EXPONENT M = *,E12.5)
00511     1518 FORMAT ( *    EXPONENT N = *,E12.5)
00511     1520 FORMAT (//50X,*EXTERNALLY APPLIED STRESS*,/50X,27(*+*),
              *        //30X,*INITIAL*,18X,*STRESS*, 42X,*NO. OF*
              *        /32X,*STRESS*,17X,*INCREMENT*,37X, *INCREMENTS*/)
00511     1524 FORMAT (15X,*SG XX*,4X,E15.5,11X,E15.5, 37X,I5)
00511     1526 FORMAT (15X,*SG YY*,4X,E15.5,11X,E15.5)
00511     1528 FORMAT (15X,*SG XY*,4X,E15.5,11X,E15.5)
00511     1530 FORMAT (//50X,*LAMINA FAILURF CRITERIA*,/50X,23(*+*),
              *        /52X,2A10,//16X,*LAYER*,18X,*LL*,23X,*TT*,25X, *LT*/)
00511     1532 FORMAT (  25X,*ULT. STRESS*)
00511     1533 FORMAT (25X,*ULT. STRAIN*//25X,*NOTE:  ALL STRAINS ARE ENGINEER*,
              *                     *ING COMPONENTS*)
00511     1534 FORMAT (/16X,I2, 4X,*TENS.*, 4X,3(E15.5,10X),/22X,*COMP.*,
              *         4X,3(E15.5,10X))
00511     1535 FORMAT (/15X,*LAYER*,20X,*QUADRATIC INTERACTION TERM (A12)*/)
00511     1536 FORMAT (16X,I2,27X,E15.5)
```

```
 1538 FORMAT (/15X,*STIFFNESS = *,E15.5)
 1540 FORMAT (//50X,*CONTROL PARAMETERS*,/50X,18(*+*)//)
 1544 FORMAT (15X,*MAX. NO. OF ITERATIONS = *,I5)
 1546 FORMAT (15X,*CONVERGENCE CRITERIA   = *,E15.5)
 1548 FORMAT (15X,*DIVERGENCE CRITERIA    = *,E15.5)
 1553 FORMAT(3X,8E14.5/3X,2E14.5//)
C
      RETURN
      END
```

```
              SUBROUTINE PROP(SG,EXX,EYY,VXY,VYX,GXY,KSGM,ILD,LAY,LPROP)
        C
        C     ROUTINE PROP COMPUTES INITIAL LAMINATE CONSTANTS (LPROP=1)
        C     OR NONLINEAR LAMINATE PROPERTIES AS FUNCTION OF RESULTANT STRESS (LPROP=2)
        C
        C                                    * VARIABLE DICTIONARY *
        C
        C     CSUM(I,J)               : THICKNESS WEIGHTED SUM OF LAYER STIFFNESSES
        C                               IN LAMINATE COORDINATES
        C     HEL                     : LONGITUDINAL YOUNG#S MODULUS OF LAMINATE
        C     HET                     : TRANSVERSE YOUNG#S MODULUS OF LAMINATE
        C     NULT                    : POISSON RATIO IN LONG.-TRANS.
        C     NUTL                    : POISSON RATIO IN TRANS.-LONG.
        C     HGLT                    : IN-PLANE SHEAR MODULUS OF LAMINATE
        C
        C     ***********************************************************************
        C
0015          DIMENSION       TNS(3,3),CMT(3,3),TMP(3,3),CSUM(3,3),
             1                EXX(KSGM),EYY(KSGM),VXY(KSGM),VYX(KSGM),GXY(KSGM)
0015          DIMENSION       T(20),SG(60,1),IANG(20)
0015          DIMENSION       H(25),TH(25),A(3,3)
0015          REAL IANG ,NULT,NUTL
0015          EQUIVALENCE        (CSUM(1),A(1))
0015          COMMON    /SET02/ TT,T,IANG
        C
        C     INITIALIZE
0015          DO 30 I=1,3
0016          DO 30 J=1,3
0017          CSUM(I,J) = 0.00
0022       30 CONTINUE
        C
0026          GO TO (50,75), LPROP
0034       50 CONTINUE
0034          T0 = TT/2.
0036          H0 = -T0
0037          TSUM = T(1)
0041          TH(1) = T(1)
0042          H(1) = H0 + T(1)
0044          DO 62 K=2,LAY
0045          TSUM = TSUM + T(K)
0047          H(K) = H0 + TSUM
0052          TH(K) = H(K)-H(K-1)
0055       62 CONTINUE
0057          GO TO 90
        C
0057       75 CONTINUE
0057          DO 85 K=1,LAY
0061          TH(K) = T(K)/TT
0064       85 CONTINUE
0066       90 CONTINUE
        C
        C     GENERATE SIMILARITY MATRIX OF CMT FOR EACH LAYER
        C                         CSIM = TNS**(-1) * CMT * TNS
0066          DO 150 K=1,LAY
```

```
'0070             CALL TRANS(TNS,K)
'0072             CALL CMATX(CMT,SG,LAY,K,LPROP)
'0103             CALL MXMULD(CMT,TNS,TMP,3,3,3,3,3,3)
'0114             CALL INVRTD(TNS,3,3,DET,1.0E-12,IRANK,1.0E-30)
'0123             CALL MXMULD(TNS,TMP,CMT,3,3,3,3,3,3)
'0134             DO 110 KI =1,3
'0141             DO 110 KJ =1,3
'0142             CSUM(KI,KJ)=CSUM(KI,KJ)+CMT(KI,KJ)*TH(K)
'0152         110 CONTINUF
'0156         150 CONTINUF
           C
           C      TNVERT RFSULTANT MATRIX
'0161             CALL INVRTD(CSUM,3,3,DET,1.0E-12,IRANK,1.0E-30)
           C
'0167             GO TO (175,500), LPROP
'0201         175 CONTINUF
'0201             HEL = 1./(A(1,1)*TT)
'0204             HET = 1./(A(2,2)*TT)
'0207             NULT = -A(1,2)/A(1,1)
'0211             NUTL = -A(1,2)/A(2,2)
'0213             HGLT = 1./(2.*A(3,3)*TT)
'0216             WRITE(6,1510)
'0222             WRITE(6,1522) HEL
'0230             WRITE(6,1524) HET
'0236             WRITE(6,1526) NULT
'0244             WRITE(6,1528) NUTL
'0252             WRITE(6,1532) HGLT
'0260             WRITE(6,1560)
'0264             RETURN
           C
'0265         500 CONTINUE
           C      COMPUTE NONLINEAR PROPERTIES AS FUNCTION OF STRESS
'0265             EXX(ILD) = 1./CSUM(1,1)
'0274             EYY(ILD) = 1./CSUM(2,2)
'0276             VXY(ILD) =-CSUM(1,2)/CSUM(1,1)
'0300             VYX(ILD) =-CSUM(1,2)/CSUM(2,2)
'0303             GXY(ILD) = 1./CSUM(3,3)
'0305             RETURN
           C
'0305        1510 FORMAT (//45X,*LAMINATE CONSTANTS (STRESS-STRAIN)*,/45X,36(*-*))
'0305        1522 FORMAT ( /48X,*EXX  = *,E15.5)
'0305        1524 FORMAT (  48X,*EYY  = *,E15.5)
 0305        1526 FORMAT (  48X,*VYX  = *,E15.5)
 0305        1528 FORMAT (  48X,*VXY  = *,E15.5)
 0305        1532 FORMAT (  48X,*GXY  = *,E15.5)
 0305        1560 FORMAT (1H1,50X,*APPLIED STRESS ANALYSIS*/50X,22(*-*))
 0305             END
```

```
                SURROUTINE TRANS(T,I)
         C
         C      ROUTINE TRANS COMPUTES TRANSFORMATION MATRIX
         C
 0005           DIMENSION        T(3,3),SIN2(20),COS2(20),SINS(20),COSS(20)
 0005           COMMON   /SET06/ SIN2,COS2,SINS,COSS
         C
 0005           T(1,1) = COSS(I)
 0006           T(2,2) = COSS(I)
 0010           T(1,2) = SINS(I)
 0011           T(2,1) = SINS(I)
 0013           T(1,3) = SIN2(I)
 0014           T(3,1) =-SIN2(I)/2.
 0017           T(3,2) = SIN2(I)/2.
 0020           T(2,3) =-SIN2(I)
 0022           T(3,3) = COSS(I)-SINS(I)
         C
 0024           RETURN
 0025           END
```

```
                 SUBROUTINE CMATX(C,SG,N,I,LPROP)
           C
           C     ROUTINE CMATX COMPUTES C MATRIX
           C
 0010            COMMON   /SET01/ E11,E22,V12,V21,G12
 0010            COMMON   /SET10/ STY,SCY,TY,XM,XN
 0010            DIMENSION        E11(20),E22(20),V12(20),V21(20),G12(20),
                1                 SCY(20),STY(20),TY(20)
 0010            DIMENSION        SG(60,1),C(3,3)
           C
 0010            DENOM = 1.0-V12(I)*V21(I)
 0014            C(1,1)   = E11(I)/DENOM
 0016            C(2,2)   = E22(I)/DENOM
 0017            IF(LPROP.EQ.1)  C(3,3)   = 2.*G12(I)
 0023            IF(LPROP.EQ.2)  C(3,3)   = G12(I)/(1.0+(SG(I+2*N,1)/TY(I))**2)
 0033            C(1,2)   = V12(I)*E22(I)/DENOM
 0037            C(2,1)   = C(1,2)
 0041            C(1,3)   = 0.0
 0042            C(2,3)   = 0.0
 0043            C(3,1)   = 0.0
 0044            C(3,2)   = 0.0
           C
 0045            RETURN
 0045            END
```

```
                 SUBROUTINE REGA1(X,Y,IPTS,OPT,A0,A1,M,IPRT)
           C
           C     ROUTINE REGA1 PERFORMS LEAST-SQUARES CURVE-FIT TO
           C     GEOMETRIC CURVE OF FORM:  Y = A0 + A1*X**M
           C
           C                                * VARIABLE DICTIONARY *
           C
           C     OPT = 1: DETERMINE A1 + M.        IPRT = 0: DO NOT PRINT RESULTS
           C         = 2: DETERMINE M                   = 1: PRINT RESULTS
           C         = 3: DETERMINE A1
           C
           C     A0: INTERCEPT   (PASSED TO SUBROUTINE)
           C     A1: COEFFICIENT (PASSED TO/OR DETERMINED BY SUBROUTINE)
           C      M: EXPONENT    (PASSED TO/OR DETERMINED BY SUBROUTINE)
           C
           C     ***************************************************************************
           C
0013             INTEGER          OPT
0013             REAL             M
0013             DIMENSION        X(IPTS),Y(IPTS),OPTION(3)
0013             EQUIVALENCE      (SX,SNUM),(SY,SDEN)
0013             DATA             OPTION/10H  A1 AND M,10H    M ONLY,
                *                       10H   A1 ONLY/
           C
           C
0013             N=IPTS
           C
           C     INITIALIZE PARTIAL SUMS TO ZERO
0014             SX  = 0.0
0015             SX2 = 0.0
0016             SY  = 0.0
0017             SY2 = 0.0
0020             SXY = 0.0
0021             NDELET = 0
           C
           C     CHOSE APPROPRIATE CURVE-FIT OPTION
0022             GO TO (50,250,500),OPT
           C
           C     FIT M AND A1
0030          50 CONTINUE
0030             DO 100 I = 1,N
           C     DELETE DATA POINTS YIELDING NEGATIVE ARGUMENTS
           C     CONVERT TO LOG FORM
0032             IF((Y(I)-A0).LE.1.E-20)  GO TO 80
0036             XP =   ALOG(X(I))
0046             YP =   ALOG(Y(I)-A0)
           C     COMPUTE INTERMEDIATE SUMS, SUM SQUARES, AND CROSS PRODUCT SUMS
0057             SX  = SX  + XP
0061             SX2 = SX2 + XP**2
0063             SY  = SY  + YP
0065             SY2 = SY2 + YP**2
0067             SXY = SXY + XP*YP
0071             GO TO 100
0072          80 CONTINUE
```

```
10072          NDELET = NDELET + 1
10074      100 CONTINUE
10077          N = N - NDELET
         C     COMPUTE REGRESSION EQUATION PARAMETERS
10100          M = (SXY - SX*SY/N)/(SX2 - SX**2/N)
10111          A1 = EXP(SY/N - M*SX/N)
         C     COMPUTE CORRELATION COEFFICIENT (ABS. VAL.)
10123          R = (N*SXY-SX*SY)/SQRT((N*SX2-SX**2)*(N*SY2-SY**2))
10143          R = ABS(R)
10144          GO TO 800
         C
         C     FIT M ONLY
10150      250 CONTINUE
10150          GO TO 800
         C
         C     FIT A1 ONLY
10151      500 CONTINUE
10151          DO 550 I=1,N
10153          YP = Y(I)-A0
10156          XP = X(I)
10160          SNUM = SNUM + YP*XP**M
10166          SDEN = SDEN + XP**(2.*M)
10175      550 CONTINUE
10177          A1 = SNUM/SDEN
10201          R = 1.
         C
         C
         C     PRINT OPTIONS
10202      800 CONTINUE
10202          IF(IPRT.EQ.0) RETURN
10204          WRITE(6,1500)
10210          WRITE(6,1510) OPTION(OPT)
10225          WRITE(6,1515) R
10233          WRITE(6,1520) A0
10245          WRITE(6,1524) A1
10257          WRITE(6,1528) M
         C
10265          IF(NDELET.NE.0)  WRITE(6,1810)  NDELET
         C
         C
10303     1500 FORMAT (1H1,////49X,*LEAST SQUARES REGRESSION ANALYSIS OF FORM*/
              *          60X,*Y = A0 + A1.X..M*////)
10303     1510 FORMAT (50X,*FIT PARAMETERS    *,A10)
10303     1515 FORMAT (50X,*CORRELATION COEFFICIENT OF LOG CURVE = *,F6.2//)
10303     1520 FORMAT (50X,*A0 = *,1PE15.5)
10303     1524 FORMAT (50X,*A1 = *,1PE15.5)
10303     1528 FORMAT (50X,* M = *,1PE15.5)
10303     1810 FORMAT (1H1,10X,*CURVE-FIT WARNING*/25X,I5,
              *          *  DATA POINTS YIELD NEGATIVE LOG ARGUMENTS AND*,
              *          * HAVE BEEN DELETED*)
10303          RETURN
10304          END
```

```
                    SUBROUTINE ANGLE(LAY,IANG)
              C
              C     ROUTINE ANGLE REDUCES ANGLES TO VALUES BETWEEN 0 AND PI/4 FOR
              C     COMPUTING SIN AND COS
              C
10005               REAL IANG,IAVAL,IANG2
10005               DIMENSION       SINS(20),COSS(20),SIN2(20),COS2(20),IANG(20)
10005               COMMON   /SET06/ SIN2,COS2,SINS,COSS
10005               DO 72 I = 1,LAY
10006               IANG2 =  2*IANG(I)
10011               ANG   =  IANG(I)
10012               ANG2  =  IANG2
10014               RAD   =  ANG /57.295779513D0
10027               RAD2  =  ANG2/57.295779513D0
10041               IAVAL = ABS(IANG(I))
10043               IF(IAVAL.EQ.0.0) GO TO 66
10045               IF(IAVAL.NE.90.0) GO TO 62
10047               SINS(I) =   COS(0.0E0)**2
10052               COSS(I) =   SIN(0.0E0)**2
10056               SIN2(I) =   SIN(0.0E0)
10062               COS2(I) = - COS(0.0E0)
10066               GO TO 72
10070            62 CONTINUE
10070               SGN = IANG(I)/IAVAL
10072               IF(IAVAL.NE.45.0) GO TO 64
10075               SIN2(I) =   COS(0.0E0)*SGN
10101               COS2(I) =   SIN(0.0E0)
10104               GO TO 68
10106            64 CONTINUE
10106               IF(IAVAL.LT.45.0)GO TO 66
10111               RDA     =  (2.*IAVAL-90.)/57.295779513
10114               SIN2(I) =  SGN* COS(SGN*RDA)
10122               COS2(I) = -SGN* SIN(SGN*RDA)
10130               GO TO 68
10132            66 CONTINUE
10132               SIN2(I) =   SIN(RAD2)
10136               COS2(I) =   COS(RAD2)
10142            68 CONTINUE
10142               SINS(I) =   SIN(RAD)**2
10146               COSS(I) =   COS(RAD)**2
10152            72 CONTINUE
10156               RETURN
10156               END
```

```
                SUBROUTINE MXMULD(A,B,C,NROWA,NCOLA,NCOLB,MA,NA,NB)            R06D0001
          C
          C     ROUTINE MXMULD MULTIPLIES TWO MATRICES (A * B), STORES RESULT IN C
          C
)0014           DIMENSION A(NROWA,NCOLA),B(NCOLA,NCOLB),C(NROWA,NCOLB)         R0
)0014           REAL          A,B,C,X
)0014           DO 20 I=1,MA
)0015           DO 20 J=1,NB                                                   R06D0005
)0016           X=0.                                                           R06D0006
)0017           DO 10 K=1,NA                                                   R06D0007
)0021        10 X=X+A(I,K)*B(K,J)                                              R06D0008
)0035        20 C(I,J)=X                                                       R06D0009
          C
)0047           RETURN                                                         R06D0010
)0047           END                                                            R06D0011
```

```
              SUBROUTINE INVRTD(A,NDIM,N,DETA,EPS,IRANK,UNDER)
        C
        C     ROUTINE INVRTD INVERTS AN N X N MATRIX USING GAUSS-JORDAN
        C     ELIMINATION METHOD
        C
        C                                   * VARIABLE DICTIONARY *
        C
        C     A(N,N)               : MATRIX TO BE INVERTED PASSED,INVERSE RETURNED
        C     NDIM                 : UPPER LIMIT TO MATRIX DIMENSION
        C     N.                   : DIMENSION OF MATRIX
        C     IRANK                : RANK OF MATRIX
        C     DETA                 : DETERINANT OF MATRIX
        C     EPS                  : ADJUSTABLE TOLERANCE FACTOR COMPARED TO
        C                            VALUE OF PIVOTAL ELEMENT DURING INVERSION
        C     UNDER                : UNDERFLOW LIMIT (CHECK ON COMPUTED VAR.)
        C
        C     **************************************************************************
        C
00012         DIMENSION A(NDIM,NDIM)
00012         INTEGER IR(60),IC(60),R,S
        C     CHECK MATRIX ELEMENTS FOR UNDERFLOW POSSIBILITIES
00012         DO 5 I=1,N
00013         DO 5 J=1,N
00014         IF( ABS(A(I,J)).LT.UNDER)  A(I,J) = 0.0E 00
00027       5 CONTINUE
00034         DETA=1.                                                           R12D0006
00035         SUM=0.                                                            R12D0007
00036         DO 10 I = 1,N                                                     R12D0008
00037         DO 10 J = 1,N                                                     R12D0009
00040      10 SUM=SUM+A(I,J)**2                                                 R12D0010
00052         SUM= SQRT(SUM)                                                    R12D0011
00054         DMA = N**2                                                        R12D0012
00061         RMS=SUM/DMA                                                       R12D0013
00063         TOL=EPS*RMS                                                       R12D0014
00064         DO 20 I = 1,N                                                     R12D0015
00066         IR(I)=0                                                           R12D0016
00067      20 IC(I)=0                                                           R12D0017
00072         S=0                                                               R12D0018
00073         R = N                                                             R12D0019
00074      30 I=0                                                               R12D0020
00075         J=0                                                               R12D0021
00076         TEST=0.0                                                          R12D0022
00077         DO 50 K = 1,N                                                     R12D0023
00100         IF(IR(K).NE.0)GO TO50                                             R12D0024
00102         DO 40 L = 1,N                                                     R12D0025
00103         IF(IC(L).NE.0)GO TO40                                             R12D0026
00105         X= ABS(A(K,L))                                                    R12D0027
00112         IF(X.LT.TEST)GO TO40                                              R12D0028
00114         I=K                                                               R12D0029
00116         J=L                                                               R12D0030
00117         TEST=X                                                            R12D0031
00120      40 CONTINUE                                                          R12D0032
00123      50 CONTINUE                                                          R12D0033
00126         PIV=A(I,J)                                                        R12D0034
```

```
      IF( ABS(DETA).LT.UNDER)  DETA = 0.00E 00
      DETA=PIV*DETA                                               R12D0035
      IF ( ABS(PIV) .LE. TOL) GO TO 150                           R12D0036
      IR(I)=J                                                     R12D0037
      IC(J)=I                                                     R12D0038
      PIV = 1.0E0/PIV                                             R12D0039
      A(I,J)=PIV                                                  R12D0040
      DO 60 K = 1,N                                               R12D0041
   60 IF(K.NE.J)A(I,K)=A(I,K)*PIV                                 R12D0042
      DO 90 K = 1,N                                               R12D0043
      IF (K.EQ.I) GO TO 90                                        R12D0044
      PIV1 = A(K,J)                                               R12D0045
   70 DO 80 L = 1,N                                               R12D0046
      IF( ABS(PIV1).LT.UNDER)  PIV1=0.0E 00
      IF( ABS(A(I,L)).LT.UNDER)  A(I,L) = 0.00E 00
   80 IF(L.NE.J)A(K,L)=A(K,L)-PIV1*A(I,L)                         R12D0047
   90 CONTINUE                                                    R12D0048
      DO 100 K = 1,N                                              R12D0049
  100 IF(K.NE.I)A(K,J)=-PIV*A(K,J)                                R12D0050
      S=S+1                                                       R12D0051
      IF(S.LT.R)GO TO 30                                          R12D0052
  110 DO 140 I = 1,N                                              R12D0053
      K=IC(I)                                                     R12D0054
      M=IR(I)                                                     R12D0055
      IF(K.EQ.I)GO TO 140                                         R12D0056
      DETA=-DETA                                                  R12D0057
      DO 120 L = 1,N                                              R12D0058
      TEMP=A(K,L)                                                 R12D0059
      A(K,L)=A(I,L)                                               R12D0060
  120 A(I,L)=TEMP                                                 R12D0061
      DO 130 L = 1,N                                              R12D0062
      TEMP=A(L,M)                                                 R12D0063
      A(L,M)=A(L,I)                                               R12D0064
  130 A(L,I)=TEMP                                                 R12D0065
      IC(M)=K                                                     R12D0066
      IR(K)=M                                                     R12D0067
  140 CONTINUE                                                    R12D0068
  150 IRANK=S                                                     R12D0069
      RETURN                                                      R12D0070
      END
```

```
                 SUBROUTINE NRTRM(LAY,SG,F,G,H,I)
           C
           C     ROUTINE NRTRM COMPUTES ELEMENTS OF DERIVATIVE MATRIX IN
           C     NEWTON-RAPHSON ANALYSIS
           C
00011            REAL              SG(60,1)
00011            REAL              E11(20),E22(20),V12(20),V21(20),G12(20),
                1                  SCY(20),STY(20),TY(20)
00011            DIMENSION         S11(20),S12(20),S21(20),S22(20),
                1                  SINS(20),COSS(20),SIN2(20),COS2(20),
                2                  F(3,20),G(3,20),H(3,20)
00011            COMMON   /SET01/  E11,E22,V12,V21,G12
00011            COMMON   /SET06/  SIN2,COS2,SINS,COSS
00011            COMMON   /SET08/  S11,S22,S12,S21
00011            COMMON   /SET10/  STY,SCY,TY,XM,XN
           C
00011            N =  LAY
00012            T12S =  (SG(I+2*N,1)/TY(I))**2
00016            T12D =   SG(I+2*N,1)/TY(I) **2
00022            T22S =  (SG(I+N,1)/TY(I))**2
00025            T22D =   SG(I+N,1)/TY(I) **2
00031            C12T =  SG(I+2*N,1)*SG(I+N,1)/TY(I)**2
00037            IF(SG(I+N,1)- 0.0) 1,2,2
00043          1 VAL =SCY(I)*1.0E-60
00046            RAT=SG(I+2*N,1)*SG(I+N,1)/SCY(I)
00054            VAL= ABS(VAL)
00056            RAT = ABS(RAT)
00057            IF( RAT.LE.VAL) RAT=0.0E00
00062            C12S = RAT/ SCY(I)
00064            TS22S  =(SG(I+N,1)/SCY(I))**2
00070            GO  TO  3
00071          2  VAL = STY(I)* 1.00E-60
00074             RAT =SG(I+2*N,1)*SG(I+N,1)/STY(I)
00102             VAL = ABS(VAL)
00104             RAT =  ABS(RAT)
00105             IF(RAT.LE.VAL) RAT=0.0E00
00110            C12S=RAT/STY(I)
00112            TS22S  =(SG(I+N,1)/STY(I))**2
00116          3  S12S = T12S +TS22S
00120            IF( ABS(S12S).LT.1.00E-20)  GO TO 40
00123            PN12 =  S12S**((XN-1.)/2.)
00132            PN32 =  S12S**((XN-3.)/2.)
00141            PM12 =  S12S**((XM-1.)/2.)
00150            PM32 =  S12S**((XM-3.)/2.)
00157            GO TO 80
00160         40 CONTINUE
00160            PN12 =  0.
00161            PN32 =  0.
00162            PM12 =  0.
00163            PM32 =  0.
00164         80 CONTINUE
           C
00164            SNS  =  SINS(I)
00166            CSS  =  COSS(I)
```

```
00167          SN2   =  SIN2(I)
00171          CS2   =  COS2(I)
          C
00172          F(1,I) =  S11(I)*CSS + S21(I)*SNS
00177          F(2,I) =  S12(I)*CSS + SNS/E22(I) + SNS/E22(I)*PN12
              1          + SNS*TS22S*(XN-1.)/E22(I)*PN32
              2          - SN2*(XM-1.)/(2.*G12(I))*PM32*C12S
00227          F(3,I) =  -SN2/(2.*G12(I)) - SN2/(2.*G12(I))*PM12
              1          - SN2*T12S*(XM-1.)/(2.*G12(I))*PM32
              2                    + C12T*(XN-1.)/E22(I)*PN32
00253          G(1,I) =  S11(I)*SNS + S21(I)*CSS
00260          G(2,I) =  SNS*S12(I) + CSS/E22(I) + PN12*CSS/E22(I)
              1          + CSS      *(XN-1.)*PN32/E22(I)*TS22S
              2          + (XM-1.)*C12S*SN2*PM32/(2.*G12(I))
00310          G(3,I) =  SN2/(2.*G12(I)) + SN2*PM12/(2.*G12(I))
              1         + SN2*(XM-1.)/(2.*G12(I))*PM32*T12S
              2        + (XN-1.)*C12T*PN32*CSS/E22(I)
00334          H(1,I) =  (S11(I)-S21(I))*SN2/2.
00343          H(2,I) =  S12(I)*SN2/2. - SN2/(2.*E22(I)) - PN12*SN2
              1         /(2.*E22(I)) - (XN-1.)*PN32*SN2/(2.*E22(I))
              2              *TS22S             +(XM-1.)*C12S*PM32
              3              *CS2/(2.*G12(I))
00373          H(3,I)=  CS2/(2.*G12(I))+ PM12*CS2/(2.*G12(I))
              1          + (XM-1.)*PM32              *T12S*CS2/(2.*G12(I))
              2          - (XN-1.)*C12T*PN32                *SN2/(2.*E22(I))
00421          RETURN
          C
00422          END
```

```
                SUBROUTINE RESET(LT3,SG,RVAL)
          C
          C     ROUTINE RESET SETS RELATIVEY SMALL STRESS TERMS EQUAL TO ZERO,
          C     TO AVOID CONVERGENCE DIFFICULTIES (I.E., VALUES ) RVAL)
          C
          C
000006          DIMENSION        SG(LT3,1)
          C
          C     FIND MAXIMUM STRESS VALUE
000006          SGMX =   ABS(SG(1,1))
000010          DO 318 K=2,LT3
000011          IF( ABS(SG(K,1)).GT.SGMX)  SGMX =   ABS(SG(K,1))
000025      318 CONTINUE
          C     COMPARE EACH RELATIVE STRESS VALUE TO RVAL
000030          DO 319 K=1,LT3
000031          RAT =   ABS(SG(K,1)/SGMX)
000037          IF(RAT.LT.RVAL)      SG(K,1)  =  0.0E 00
000046      319 CONTINUE
000051          RETURN
000051          END
```

```
                   SUBROUTINE CONVR(LAY,SG,SG1,KSG,IRTN)
              C
              C    ROUTINE CONVR CHECKS FOR CONVERGED SOLUTION DURING NEWTON-RAPHSON
              C    ANALYSIS.  ALSO CHECKS ITERATION LIMIT AND DIVERGENCE LIMIT.
              C
000010             DIMENSION      SG(60,1),SG1(60,1),DIF(60)
000010             COMMON   /SET11/ EPS,UPBD,NIT,IT,SMLT
              C
000010             ICON =  1
000011             N =  LAY
000012             LT3 =  LAY*3
000013             IRTN=1
              C
              C    CONVERGENCE  CHECK
000014             DO 375 J3=1,LT3
000016             SUB =   ABS(SG(J3,1))- ABS(SG1(J3,1))
000024             IF(SG1(J3,1).EQ.0.0E0) GO TO 330
000026             DIF(J3) =    ABS(SUB/SG1(J3,1))
000031             GO TO 335
000032         330 CONTINUE
000032             DIF(J3) =  SUB
000034         335 CONTINUE
000034             IF(DIF(J3).GT.EPS) GO TO 340
000040             GO TO 375
              C
              C    ITERATION  CHECK
000040         340 CONTINUE
000040             IF((NIT-IT).NE.0)  GO TO 350
000042             ICON =  3
000043             IF(DIF(J3).LE.UPBD)  GO TO 375
000046             ICON =  4
000047             GO TO 375
              C
              C    DIVERGENCE  CHECK
000050         350 IF(DIF(J3).LE.UPBD) GO TO 370
000053             ICON =  4
000054             GO TO 375
              C
              C
000055         370 CONTINUE
000055             ICON =  2
000056         375 CONTINUE
              C
000061             GO TO (500,400,382,386),ICON
              C
              C    NON-CONVERGENCE  DUMP
000071         382 CONTINUE
000071             WRITE(6,1720)
000075             WRITE(6,1722) EPS
000103             GO TO 395
000107         386 CONTINUE
000107             WRITE(6,1730)
000113             WRITE(6,1722) UPBD
000121         395 CONTINUE
```

```
000121          NITP =  NIT - 1
000123          WRITE(6,1741) NIT,NITP
000133          WRITE(6,1742)
000137          DO 397 I=1,LAY
000144          WRITE(6,1550) I,SG(I,1),SG(I+N,1),SG(I+2*N,1),SG1(I,1),SG1(I+N,1),
               1              SG1(I+2*N,1),DIF(I),DIF(I+N),DIF(I+2*N)
000241      397 CONTINUE
000247          IRTN=3
000250          RETURN
000250      400 IRTN=2
000251      500 RETURN
          C
          C
000252     1550 FORMAT (I4,1X,2(3E13.5,4X),3E13.5)
000252     1720 FORMAT (* SOLUTION  FOR  STRESS  DOES  NOT  CONVERGE.*)
000252     1722 FORMAT (* RELATIVE  ERROR  .GT.*,E15.5)
000252     1730 FORMAT (* SOLUTION  FOR  STRESS  DIVERGES.*)
000252     1741 FORMAT (/18X,*(ITERATION *,I3,*)*,28X,*(ITERATION *,I3,*)*)
000252     1742 FORMAT (  * LAYER*,4X,*SGM X*,8X,*SGM Y*,8X,*SGM XY*,11X,*SGM X*,
               1        8X,*SGM Y*,8X,*SGM XY*,11X,*REL X*,8X,*REL Y*,8X,
               2        *REL XY*/)
000252          END
```

```
                SUBROUTINE LAMTST(LAY,SG,SGS,EPN,PS,KSG,KSGM,IFCN,UFAIL,FAC,SW)
          C
          C     ROUTINE LAMTST PERFORMS FAILURE ANALYSIS,  STIFFNESS TEST, ULTIMATE
          C     STRESS, ULTIMATE STRAIN, AND QUADRATIC INTERACTION
          C
          C                                        * VARIABLE DICTIONARY *
          C
          C     UFAIL(I)                : INDICATE FAILURE UNDER SEPARTE MODES
          C                               FOR MULTI-MODE FAILURE ANALYSIS (I.E., IFCN=4)
          C     IFCN                    : FAILURE ANALYSIS OPTION
          C     LFAIL                   : FAILURE MODE
          C     IJJ                     : ORIEINATION OF STRESS FAILURE
          C     KJJ                     : ORIENTATION OF STRAIN FAILURE
          C     QIT(I)                  : QUADRATIC INTERACTION TERM FOR LAYER I
          C     QPV(I)                  : QUADRATIC INTERACTION TERM FOR LAYER I  PROV.LOAD
          C     FAC                     : INTERPOLATION FACTOR
          C     IST                     : STORE LAYER NO. AT FAILURE
          C     ISV                     : STORE STRESS ORIENTATION NO. AT FAILURE
          C     KSV                     : STORE STRAIN ORIENTATION NO. AT FAILURE
          C
          C     ************************************************************************
          C
000016          DIMENSION       SG(60,1),SGS(60)
000016          DIMENSION       EPS11(20),EPS22(20),EPS12(20)
000016          DIMENSION       P11(20),P22(20),P12(20),
               1                ULT(6,2),ULTIMA(6,2,20),
               1                EP11(20),EP22(20),EP12(20)
000016          DIMENSION       QIT(20),QPV(20)
000016          DIMENSION       A11(20),A22(20),A44(20),A12(20),B1(20),B2(20)
000016          DIMENSION       EPN(60,1),PS(60)
000016          INTEGER         UFAIL(3),SW,T
          C
000016          COMMON   /SET03/EP11,EP22,EP12
000016          COMMON   /SET04/ SO11,SO22,SO12,SM11,SM22,SM12
000016          COMMON   /SET05/ ULTIMA,STIFF
000016          COMMON   /SET07/EPS11,EPS22,EPS12
000016          COMMON   /SET14/ A11,A22,A44,A12,B1,B2
          C
          C     EVALUATE QUADRATIC INTERACTION COEFFICIENTS IF QUAD. INTER. FAILURE
000016          IF((IFCN.EQ.3).OR.(IFCN.EQ.4))   CALL QUADCF(LAY)
000032          IST = 1
000033          LFAIL = 0
000034          IPT = 1
          C
000035      500 CONTINUE
000035          DO 675 I=IST,LAY
000037          T = I
          C
000040          DO 550 II=1,6
000042          DO 550 JJ=1,2
000043          ULT(II,JJ) = ULTIMA(II,JJ,I)
000052      550 CONTINUE
          C
          C     TEST  1:  STIFFNESS  TEST
```

```
000056          IF(KSG.EQ.1) GO TO 560
000060          IF(SO11.EQ.0.0E0)  GO TO 560
000061          RATIO =  ABS(SM11/(EP11(I)-EPS11(I)))
000065          IF(RATIO.LT.STIFF) GO TO 677
         C      TEST  2:  ULTIMATE STRESS
         C                ULT(1,1):  MAX. AXIAL TENS.       ULT(1,2):  MAX. AXIAL COMP.
         C                ULT(2,1):  MAX. TRAN. TENS.       ULT(2,2):  MAX. TRAN. COMP.
         C                ULT(3,1):  MAX. SHEAR
000070     560 CONTINUE
000070          IF(.NOT.((IFCN.EQ.1).OR.(IFCN.EQ.4)))  GO TO 570
000100          IF((UFAIL(1).EQ.1).AND.(IFCN.EQ.4))  GO TO 570
000107          IJJ=1
000110          IF(SG(I,1)-0.0) 1,1,2
000113       1 IF( ABS(SG(I,1))-ULT(1,2))3,679,679
000120       2 IF(      SG(I,1) -ULT(1,1))3,679,679
000124       3 K= I+LAY
000126          IJJ=2
000127          IF(SG(K,1)-0.0)4,4,5
000132       4 IF( ABS(SG(K,1))-ULT(2,2))6,679,679
000137       5 IF(SG(K,1) -ULT(2,1))6,679,679
000143       6 K= I+2*LAY
000145          IJJ=3
000146          IF( ABS(SG(K,1))-ULT(3,1))570,679,679
         C
         C      TEST  3:  ULTIMATE STRAIN
         C                ULT(4,1):  MAX. AXIAL TENS.       ULT(4,2):  MAX. AXIAL COMP.
         C                ULT(5,1):  MAX. TRAN. TENS.       ULT(5,2):  MAX. TRAN. COMP.
         C                ULT(6,1):  MAX. SHEAR
000153     570 CONTINUE
000153          IF(.NOT.((IFCN.EQ.2).OR.(IFCN.EQ.4)))  GO TO 580
000163          IF((UFAIL(2).EQ.1).AND.(IFCN.EQ.4))  GO TO 580
000173          KJJ = 1
000174          IF(EPN(I,1)-0.0) 71,71,72
000177      71 IF( ABS(EPN(I,1))-ULT(4,2))73,689,689
000204      72 IF(      EPN(I,1)-ULT(4,1))73,689,689
000210      73 CONTINUE
000210          K = I+LAY
000212          KJJ = 2
000213          IF(EPN(K,1)-0.0) 74,74,75
000216      74 IF( ABS(EPN(K,1))-ULT(5,2))76,689,689
000223      75 IF(EPN(K,1)-ULT(5,1))76,689,689
000227      76 CONTINUE
000227          K = I+2*LAY
000231          KJJ = 3
000232          IF( ABS(EPN(K,1))-ULT(6,1))580,689,689
         C
         C
         C      TEST  4:  QUADRATIC INTERACTION
000237     580 CONTINUE
000237          IF(.NOT.((IFCN.EQ.3).OR.(IFCN.EQ.4)))  GO TO 590
000247          IF((UFAIL(3).EQ.1).AND.(IFCN.EQ.4))  GO TO 675
000257          QIT(I)  = A11(I)*SG(I,1)**2 + A22(I)*SG(I+LAY,1)**2
               1          +A44(I)*SG(I+2*LAY,1)**2 + A12(I)*SG(I,1)*SG(I+LAY,1)
               2          +B1(I)*SG(I,1) + B2(I)*SG(I+LAY,1)
000312          IF(      QIT(I) .GT.1.0)  GO TO 699
```

```
                C
000316            590 CONTINUE
                C
000316            675 CONTINUE
000321                IF (LFAIL.NE.0)  GO TO 810
000322                WRITE(6,1995)
000325                RETURN
                C
                C       SET INDICATOR OF FAILURE MODE
000326            677 LFAIL = 1
000327                KSG = KSGM
000334                GO TO 700
000335            679 LFAIL = 2
000336                IST = T
000340                GO TO 700
000340            689 LFAIL = 3
000341                IST = T
000343                GO TO 700
000343            699 LFAIL = 4
000344                IST = T
000346                QPV(T) = A11(T)*SGS(T) **2 + A22(T)*SGS(T+LAY) **2
                   *         +A44(T)*SGS(T+2*LAY )**2 + A12(T)*SGS(T )*SGS(T+LAY)
                   *         +B1(T)*SGS(T ) + B2(T)*SGS(T+LAY )
                C
                C       PRINT FAILURE MODE
000401            700 CONTINUE
000401                IF(IPT.EQ.1)  WRITE(6,1990)
000413                IPT = 0
000414                IF(IFCN.NE.4)  KSG=KSGM
000420                GO TO (701,703,723,743), LFAIL
000430            701 WRITE(6,1450)
000434                GO TO 850
000440            703 CONTINUE
000440                ISV = IJJ
000442                GO TO(704,705,706),IJJ
000450            704 WRITE(6,1452)
000454                GO TO 799
000460            705 WRITE(6,1453)
000464                GO TO 799
000470            706 WRITE(6,1454)
000474                GO TO 799
                C
000500            723 CONTINUE
000500                KSV = KJJ
000502                GO TO (724,725,726), KJJ
000510            724 WRITE(6,1462)
000514                GO TO 799
000520            725 WRITE(6,1463)
000524                GO TO 799
000530            726 WRITE(6,1464)
000534                GO TO 799
                C
000540            743 WRITE(6,1472)  QIT(T),T,QPV(T),T
                C
000554            799 CONTINUE
```

```
000554           IF(IFCN.NE.4)   GO TO 850
000562           LFM = LFAIL-1
000564           UFAIL(LFM) = 1
000566           IF((UFAIL(1).EQ.1).AND.(UFAIL(2).EQ.1).AND.(UFAIL(3).EQ.1))
                1                                                          KSG = KSGM
000605           GO TO 500
000606       810 CONTINUE
000606           IF (KSG.EQ.1)   GO TO 850
000610           SW = 1
000611           IF( LFAIL.EQ.2 )        FAC = SINT(SG,SGS,ISV,1,IST,LAY)
000624           IF( LFAIL.EQ.3 )        FAC = SINT(EPN,PS,KSV,2,IST,LAY)
000637           IF( LFAIL.EQ.4 )        FAC = (1.0-QPV(IST))/(QIT(IST)-QPV(IST))
000646       850 WRITE(6,1495)
           C
000652      1450 FORMAT (//* LAMINATE  HAS  FAILED+   STIFFNESS  TEST  FAILURE*)
000652      1452 FORMAT (//* LAMINATE  HAS  FAILED+   SG 11  EXCEEDS  MAXIMUM*)
000652      1453 FORMAT (//* LAMINATE  HAS  FAILED+   SG 22  EXCEEDS  MAXIMUM*)
000652      1454 FORMAT (//* LAMINATE  HAS  FAILED+   SG 12  EXCEEDS  MAXIMUM*)
000652      1462 FORMAT (//* LAMINATE  HAS  FAILED+   EP 11  EXCEEDS  MAXIMUM*)
000652      1463 FORMAT (//* LAMINATE  HAS  FAILED+   EP 22  EXCEEDS  MAXIMUM*)
000652      1464 FORMAT (//* LAMINATE  HAS  FAILED+   EP 12  EXCEEDS  MAXIMUM*)
000652      1472 FORMAT (//* LAMINATE  HAS  FAILED+   QUADRATIC INTERACTION*,
                *          * FAILURE*/27X,*QUADRATIC = *,F7.4,* FOR LAYER *,I2/
                *                     27X,*QUADRATIC = *,F7.4,* FOR LAYER *,I2,
                *          * OF PREVIOUS LOAD*)
000652      1495 FORMAT (/* AT  FIRST  POST-FAILURE  LOAD  POINT*)
000652      1990 FORMAT (////)
000652      1995 FORMAT (1H0)
           C
000652           RETURN
000653           END
```

```
                 SUBROUTINE QUADCF(LAY)
           C
           C     COMPUTES QUADRATIC FAILURE CRITERIA COEFFICIENTS
           C
000003           DIMENSION        ULTIMA(6,2,20)
000003           DIMENSION        A11(20),A22(20),A44(20),A12(20),B1(20),B2(20)
000003           COMMON    /SET05/ ULTIMA,STIFF
000003           COMMON    /SET14/ A11,A22,A44,A12,B1,B2
           C
000003           DO 100 IMTALY=1,LAY
000005           A11(IMTALY) = 1./(ULTIMA(1,1,IMTALY)*ULTIMA(1,2,IMTALY))
000013           A22(IMTALY) = 1./(ULTIMA(2,1,IMTALY)*ULTIMA(2,2,IMTALY))
000017           A44(IMTALY) = (1./ULTIMA(3,1,IMTALY))**2
000023           B1(IMTALY)  = 1./ULTIMA(1,1,IMTALY) -  1./ULTIMA(1,2,IMTALY)
000027           B2(IMTALY)  = 1./ULTIMA(2,1,IMTALY) -  1./ULTIMA(2,2,IMTALY)
000034       100 CONTINUE
           C
000037           RETURN
000037           END
```

```
                 SUBROUTINE LAYSUB (A12,STY,SCY,TY,EPS11,EPS22,EPS12)
000012           DIMENSION E11(20),E22(20),V12(20),V21(20),G12(20),A12(20),
               1    STY(20),SCY(20),TY(20),EPS11(20),EPS22(20),EPS12(20)
000012           DIMENSION  TE11(20),TE22(20),TG12(20),TV12(20),TA12(20),TSTY(20),
               1                TSCY(20),TTY(20)
000012           COMMON    /SET01/ E11,E22,V12,V21,G12
000012           COMMON    /SET17/ TE11,TE22,TG12,TV12,TA12,TSTY,TSCY,TTY
000012           DO 10 I = 1,20
000013           E11(I) = TE11(I)
000015           E22(I) = TE22(I)
000017           G12(I) = TG12(I)
000021           V12(I) = TV12(I)
000023           A12(I) = TA12(I)
000025           STY(I) = TSTY(I)
000027           SCY(I) = TSCY(I)
000031        10 TY(I)  = TTY(I)
000035           RETURN
000036           END
```

```
RUN VERSION 2.3 --PSR LEVEL 363--                                                               07/23/74

                    REAL FUNCTION SINT(VAL,PREV,IOR,SS,I,N)
              C
              C     FUNCTION SUBPROGRAM SINT DERTIMINES INTERPOLATION FACTOR FOR
              C     STRES OR STRAIN FAILURE
              C
              C                                  * VARIABLE DICTIONARY *
              C
              C     VAL(I,1)             : VALUE OF STRESS OR STRAIN AFTER FAILURE
              C     PREV(I)              : VALUE OF STRESS OR STRAUN BEFORE FAILURE
              C     FLEMT                : ELEMENT OF STRESS OR STRAIN ARRAY WHICH HAS FAIL
              C     SINT                 : INTERPOLATION FACTOR
              C
              C     ************************************************************************
              C
000011              DIMENSION        VAL(60,1),PREV(60),ULT(6,2,20)
000011              COMMON   /SET05/ ULT,STIFF
000011              INTEGER          SS,ELEMT
              C
              C
000011              IF (SS.EQ.1)  IOS=IOR
000014              IF (SS.EQ.2)  IOS=IOR+3
000020              FLEMT = (IOR-1)*N + I
              C
000023              ISN=2
000024              IF (VAL(FLEMT,1).GT.0.00)  ISN=1
              C
000030              DPRV =  ABS(PREV(ELEMT))
000033              DDIF =  ABS(VAL(ELEMT,1)-PREV(ELEMT))
000036              SINT  = (ULT(IOS,ISN,I)-DPRV)/DDIF
              C
              C
000045              RETURN
000046              END
```

```
                 BLOCK DATA
          C
          C      BLOCK DATA SUBPROGRAM INITIALIZES VARIABLES
          C      IN COMMON SETS 01,05,10,11,14
          C
000002           COMMON    /SET01/ DUM01(100)
000002           COMMON    /SET05/ ULT(240),STIFF
000002           COMMON    /SET10/ DUM10(62)
000002           COMMON    /SET11/ EPS,UPBD,NIT,IT,SMLT
000002           COMMON    /SET14/ A11,A22,A44,A12,B1,B2
000002           COMMON    /SET15/ POINTS,IPRINT,IOPT,IPTS,LUP
000002           DIMENSION         A11(20),A22(20),A44(20),B1(20),B2(20)
000002           DIMENSION         A12(20),POINTS(50,6,20)
000002           DATA              STIFF/0.100/
000002           DATA              A12/20*0.0/
000002           DATA              DUM01/100*0.0/
000002           DATA              ULT/240*0.0/
000002           DATA              DUM10/60*0.00,2*3.00/
000002           DATA              EPS/.001/,UPBD/20000./
000002           DATA              IT/100/
000002           DATA              IPRINT/0/
          C
000002           END
```

REFERENCES

(1) Z. Hashin, D. Bagchi and B. W. Rosen, "Non-Linear Behaviour of Fiber Composite Laminates", NASA CR-2313, 1973.

(2) Z. Hashin and B. W. Rosen, "the Elastic Moduli of Fiber Reinforced Materials", J. Applied Mechanics, Vol. 31, p. 223, 1964.

(3) B. W. Rosen and Z. Hashin, "Effective Thermal Expansion Coefficients and Specific Heats of Composite Materials", Int. J. Engineering Science, Vol. 8, p. 157, 1970.

NASA-Langley, 1975